OPTIMAL SHAPE DESIGN FOR A LAYERED PERIODIC STRUCTURE

A Dissertation

by

MICHAEL BRADY FLANAGAN

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

December 2002

Major Subject: Mathematics

OPTIMAL SHAPE DESIGN FOR A LAYERED PERIODIC STRUCTURE

A Dissertation

by

MICHAEL BRADY FLANAGAN

Submitted to Texas A&M University
in partial fulfillment of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

Approved as to style and content by:

David Dobson (Chair of Committee)	
Joe Pasciak (Member)	Jianxin Zhou (Member)
Vivek Sarin (Member)	Al Boggess (Head of Department)

December 2002

Major Subject: Mathematics

ABSTRACT

Optimal Shape Design for a Layered Periodic Structure. (December 2002)

Michael Brady Flanagan, B.S., Montana State University;

M.S., University of Alaska-Fairbanks

Chair of Advisory Committee: Dr. David Dobson

A multi-layered periodic structure is investigated for optimal shape design in diffraction gratings. A periodic dielectric material is used as the scattering profile for a planar incident wave. Designing optimal profiles for scattering is a type of inverse problem. The ability to fabricate such materials on the order of the wavelength of the incoming light is key for design strategies. We compute a finite element approximation on a variational setup of the forward problem. On the inverse and optimal design problem, we discuss the stability of the designs and develop computational strategies based on a level-set evolutionary approach.

To Dad, Mom, Kevin, Libby, Mathew, and my beloved Bomma.

ACKNOWLEDGMENTS

I wish to acknowledge my advisor, Dr. Dobson. You have shown great patience, and I truly appreciate all the advice and discussions. Thank you. I would like to acknowledge Joe Pasciak and Raytcho Lazarov for the insightful discussions. Life in grad school would not have gone by so seamlessy if it were not for Monique Stewart. You are a great friend and miracle worker! Thank you for not giving me morning duties! I wish to thank the following people for their undying support: Troy Henderson, April Judd, Maggie Arnold, Dave Arnold III, Dimitrije Kostic, Paul Dostert, Laura Douglass, and Thong Le. You have always made me feel welcome. For my officemate, I wish to acknowledge Stephen Shauger, for he has been a good friend. For Amir Hussain, I wish to thank you for your discussions and many fine trips out of B/CS. Hang in there, man. I wish to acknowledge Tatevik and Haik Ambartsoumian. You are truly my buddies. Finally, I wish to thank Svenja Lowitzsch, for you have always been there when I needed a hug. I love you all.

LIST OF SYMBOLS

SYMBOL Page

SYMBOL Page

TABLE OF CONTENTS

CHAPTER Page

LIST OF TABLES

LIST OF FIGURES

FIGURE Page

CHAPTER I

INTRODUCTION

A. Background

In diffractive optics much interest has been focused on periodic structures. Important materials exhibit periodicity naturally, such as crystals. But other materials can be fabricated to exhibit periodicity by repeated application of a process. The use of such structures is seen in electromagnetic applications such as spectroscopy, solid state physics, x-ray technology, photonics and optical communication devices. Specifically, diffraction gratings in 2 and 3 dimensions have been extensively studied. Much progress has been done and technology has benefited. In integrated optics, for example, such structures are used for beam focusing and splitting, optical switches, filters and more [24]. In the design of grating structures, manufacturing technology has advanced far enough to create structures on the order of the wavelength of light and smaller. See [24, 28] for an exposition of the various fabrication processes. Due to the repetitive nature of crystals and machining tools, periodic profiles are natural to study. Further studies of grating structures today include multi-layered components. Diffractive multi-layered structures can be found, for example, in the use of x-ray reflectors and beam focusing [28, 26]. These structures often possess more interesting diffractive features than single layer counterparts. For example, reflectance can be achieved at levels which are otherwise observed only through metallic reflective structures. The disadvantage with metallic objects, though, is the absorption of energy. This feature is not an issue in dielectric structures. The use of multi-layered films

---

The journal model used for this dissertation is SIAM Journal on Applied Mathematics.

dates as far back as optical technology, but rigorous studies of the structures did not start to develop until the 50's [6, 17]. In photonic crystals, multi-layered structures are used in connection with photonic band gaps [18, 15]. As one consequence of band gaps, Kosaka *et. al.*, in 1998 [20], successfully demonstrated a so-called superprism in photonic crystals achieved by a multi-layered structure, where the incident angle was relatively small compared with the propagating angle of dispersion. The implications suggested in the paper are that much smaller devices can be designed to effect sharp angle diffraction. Future technologies will utilize this and other multi-layered phenomena as their behavior is better understood.

In designing multi-layered structures, work has been devoted to looking at plane layered structures (vertically stratified) or etched layers on top of plane layered substrates [16, 7, 41]. In certain design paradigms as seen in the papers by Botten *et. al.* [7] and Maystre [25], finitely layered grating stacks are analyzed. These particular stacks are composed of a regularly periodic (x_1-direction) array of rods of varying positions and radii. This is typical of the various geometries used for analysis on the scattering problem. With the growing demand for optical systems, optimal design of structures is sought, specifically when it can reduce the cost of fabrication. The sophistication and complexity of grating structures in nanolithography have demanded precise control of the depth and period of the materials being etched. In the paper by Lim [22] a Bragg-grating is constructed to have a period of approximately 240nm. The error in the precision must be less than .1nm. This makes .04% tolerance. In the literature, there has been limited use of optimization in the models. And in such cases, only simple structures with 1 or 2 parameters are used, for example, the radii and position of a collection of rods in the grating stack. See [41] for details on this design strategy. At issue is the inherent difficulty in solving the Maxwell equations.

In the next section we will set up the problem definition and reserve careful

attention for the later chapters.

B. Problem Overview

Consider a lattice structure of dielectric material configured so that its optical properties are constant and of infinite extent in the x_3-direction (see Figure 1). It is periodic in the x_1-direction and repeats itself in a layered manner in the x_2-direction. The number of layers is finite and is denoted L. Each layer has depth d. The x_1-periodicity is scaled to 2π.

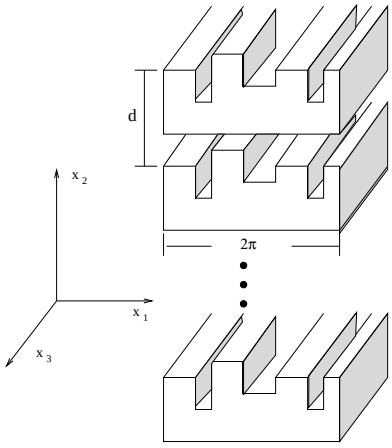


Fig. 1. L layered diffractive grating.

By passing a plane wave of incident light with the E-vector oriented parallel to the x_3 axis we may reduce this problem to a 2-D case. This is referred as E-polarized light. The optical phenomena of scattering and refraction preserve the E-polarization of light all through the material. Referring to Figure 1 the region extends in the x_1-direction by 2π. In the x_2-direction, we consider the top interface, $\Gamma_0 = \{x_2 = 0\}$. The structure extends until $\Gamma_L = \{x_2 = -Ld\}$. For future reference we will denote

these interfaces as $y_0 = 0$, and $y_1 = -Ld$. Due to the repetition, all the information about the structure can be described in the first layer, Ω_0. Define $\Omega = \bigcup_{i=0}^{i=L-1} \Omega_i$. Thus, we consider the 2-D domain as follows (see Figure 2):

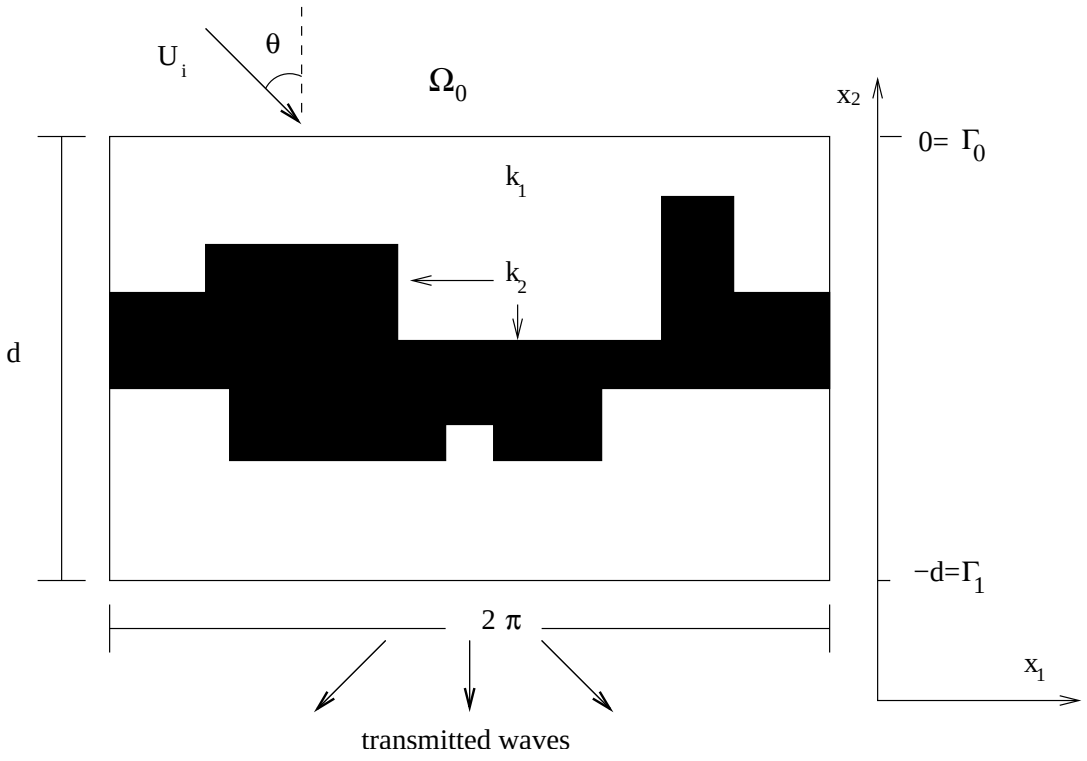


Fig. 2. First layer of domain.

The domain Ω is composed of a periodic layered dielectric medium that can be described by $\epsilon(x_1, x_2)$, the electric permittivity in the structure. The 2π periodicity in the x_1-direction means

$$\epsilon(x_1 + 2\pi m, x_2) = \epsilon(x_1, x_2), \quad \text{for all integers } m, \text{ and all } x_1 \in \mathbb{R}.$$

Because of the layering, we also have

$$\epsilon(x_1, x_2 - jd) = \epsilon(x_1, x_2), \text{ for } j = 1, \ldots, L-1, \text{ and for } -d \leq x_2 \leq 0.$$

As depicted in Figure 2 a plane wave, $e^{i(\alpha x_1 + \beta x_2 - \omega t)}$, is incident on the domain Ω. The light is characterized by its frequency, ω, and its orientation. Its angle of incidence is θ. Let $\alpha \equiv \sin\theta$. The electric permittivity ϵ used in the governing Maxwell equations defines the medium. Also present is the magnetic permeability, μ. Most optical media are nonmagnetic, which implies $\mu = 1$. From Maxwell's equations (see [35]) the problem can be reduced to a 2-D domain. E-polarized light has components $E = \langle 0, 0, V \rangle$.

$$\bigtriangledown \times \mathbf{H} = \epsilon \frac{\partial \mathbf{E}}{\partial t}, \tag{1.1}$$

$$-\bigtriangledown \times \mathbf{E} = \mu \frac{\partial \mathbf{H}}{\partial t}. \tag{1.2}$$

By applying the time derivative to (1.1), and the curl on (1.2), then combining together, the components of the vector field equations reduce to

$$\langle \partial x_1 \partial x_3 V = 0, \quad \partial x_2 \partial x_3 V = 0, \quad \partial^2 x_1 V + \partial^2 x_2 V = \epsilon\mu \frac{\partial^2 V}{\partial t^2} \rangle$$

Thus, we have the time dependent wave equation,

$$\Delta V = \epsilon\mu \frac{\partial^2 V}{\partial t^2}.$$

We define $k(x_1, x_2)^2 = \epsilon(x_1, x_2)\mu$. Assuming time dependence $V(x, t) = U(x)e^{i\omega t}$, the wave equation simplifies to the following Helmholtz equation:

$$\Delta U + k^2\omega^2 U = 0, \qquad \text{in } \mathbb{R}^2.$$

Plane waves $e^{i(k \cdot x)}$ are distinguished as incoming and outgoing relative to the

scattering profile, as directed along the oriented **k**-vector, $\mathbf{k} = k\langle\alpha, -\cos\theta\rangle$. (More on incoming and outgoing waves will be covered in Chapter II, after evanescent waves are introduced.) The periodicity of the coefficient $k(x_1, x_2)$ allows one to restrict to a periodic domain. By considering $\Omega' = \mathbb{R}^2/\{\mathbb{Z} \times \{0\}\}$ and relabeling to Ω, the problem we will study is a modified Helmholtz equation (described ahead in (1.4)) for which periodic solutions (in the x_1-direction), U_α, are sought. Plane waves in a homogeneous medium $\Omega \in \mathbb{R}^2$ all have the functional form, $Ce^{i\alpha x_1 + i\beta x_2}$. Along vertical line segments (holding x_1 fixed) the solution is a composition of traveling waves, $ce^{i\beta x_2}$. Using the Fourier decomposition and the plane wave structure, the solution U along a horizontal interface is described by

$$e^{i\alpha x_1} U_\alpha|_{\Gamma_j} = \sum_n c_n e^{i\beta_n x_2} e^{i(n+\alpha)x_1},$$

where x_2 is fixed at one of the interfaces Γ_j, $j \in \{0, L\}$. Upon substitution into the Helmholtz equation, $\Delta U + k^2\omega^2 U = 0$, the following relation is established:

$$\beta_n^2 = k^2\omega^2 - (n+\alpha)^2. \tag{1.3}$$

This defines the modes of reflection and transmission.

Recall: The k in the above formula is a constant along the horizontal interfaces characterizing the homogeneous region for which these reflection and transmission modes apply. In the next chapter, this relation will be utilized more fully.

We are now in a position to describe the modified Helmholtz equation and boundary conditions (BC's): We seek a x_1-periodic solution, $U_\alpha \equiv u$ that satisfies

$$\Delta_\alpha u + k^2\omega^2 u = 0, \qquad \text{in } \Omega, \tag{1.4}$$

$$\frac{\partial u}{\partial x_2} - T_0 u = -2i\beta_0, \quad \text{on } \Gamma_0, \tag{1.5}$$

$$\frac{\partial u}{\partial x_2} + T_L u = 0, \qquad \text{on } \Gamma_L. \tag{1.6}$$

where we use the Dirichlet-Neumann operator, T_j, defined by

$$(T_j f)(x_1) = \sum_{n \in Z} i\beta_n^j f_n e^{inx_1}, \qquad j \in \{0, L\}.$$

The f_n are the Fourier coefficients of f, and β_n^j are the modes of the scattered wave, defined ahead in 1.3. For the incoming wave, $e^{i\beta_0 x_2}$, the coefficient β_0 is given by $\sqrt{k^2\omega^2 - \alpha^2}$. The operator Δ_α is defined by $\Delta + 2i\alpha\dfrac{\partial}{\partial_1} - |\alpha|^2$. The Dirichlet-Neumann operators are used to replace the familiar Sommerfeld radiation condition. They enforce bounded outgoing waves in their formulation. These will be considered in more detail in Chapter II. The boundaries, Γ_0 and Γ_L, are intentionally positioned in a homogeneous region where no scattering occurs. This guarantees analytic behavior at these interfaces. The periodicity of the coefficients yield periodicity in the solution. From Floquet theory [14] we can seek a periodic solution U_α which is related to the full solution U by $U_\alpha = e^{-i\alpha x_1}U$. The operator Δ_α referred to in (1.4) comes from expanding the x_1-periodic solution into the original Helmholtz equation. Consider $U = e^{i\alpha x_1}U_\alpha$. To enhance readability, let $\hat{U} = U_\alpha$. Then

$$\frac{\partial^2 U}{\partial {x_1}^2} + \frac{\partial^2 U}{\partial {x_2}^2} + k^2\omega^2 U = 0.$$

The expansion,

$$e^{ix\alpha}(\hat{U}_{xx} + \hat{U}_{yy} + 2i\alpha\hat{U}_x - |\alpha|^2\hat{U} + k^2\omega^2\hat{U}) = 0$$

defines the new operator. As a side note, an alternate, more compact form of the operator Δ_α is given by

$$\Delta_\alpha \equiv (\nabla + i\alpha) \cdot (\overline{\nabla - i\alpha}) = \Delta + 2i\alpha\partial_1 - |\alpha|^2.$$

After obtaining the solution U_α, which will be shown (2.6) to lie in the space

$H^2(\Omega)$, we extract the coefficients of the outgoing waves. These particular waves include reflection and transmission. From Figure 2, the transmitted waves at the bottom of the interface suggest these waves will continue to interact with the next layer in the structure. This will create new scattered waves entering the domain from below. The scattering properties of light include both reflection and transmission at each interface in the medium. All interfaces need to be considered simultaneously for a proper picture of the wave profile. What is important is that the final modes of transmission and reflection at the bottom and top of the interface are computable. In this work a Ritz-Galerkin method [8] is employed using finite elements to model the approximate solutions. The outgoing waves are computed by analyzing the wave profile along the interfaces. This is obtained by knowing apriori the nature of the analytic periodic solutions along the homogeneous region of the top and bottom interfaces. Specifically, the solutions are composed of a linear combination of plane waves. Along each interface, let U_α be decomposed into Fourier series:

$$U_\alpha|_{\Gamma_j} = \sum_n u_n^j e^{inx_1},$$

where $u_n^j = \frac{1}{2\pi}\int_{\Gamma_j} U_\alpha(t, x_2)e^{-int}dt$.

The above Fourier decomposition will be used often in the remainder of this thesis.

C. Previous Work

The structure of this problem has been studied extensively by Dobson [10, 11, 12] for use in designing a minimal reflection profile. Much of the problem description can be found in two papers, [10, 11]. In that work the region under study was an interface profile dividing a region between two mediums. The result of the work showed the

existence of interface profiles, but it did have one drawback from an applications point of view, in that the interfaces were mixtures of the dielectric medium. The profiles were allowed to vary continuously between index of refraction k_1 to k_2. This technique is called 'relaxing' the problem [19]. In this work we will allow for a repeated multi-layered structure and will try to create distinguishable interface profiles. This will be much more attractive from a fabrication point of view for further development in optical applications and research.

D. Inverse Problem

The previous description sets the background for the essential feature of this thesis, the energy distribution of the reflected and transmitted modes of the light. The following characterizes the reflection and transmission coefficients,

$$U_\alpha \mid_{\Gamma_0} = \sum_{n\in\mathbb{Z}} r_n e^{inx+i\beta_n^0 y_0} + e^{-i\beta^0 y_0}, \tag{1.7}$$

$$U_\alpha \mid_{\Gamma_L} = \sum_{n\in\mathbb{Z}} t_n e^{inx-i\beta_n^1 y_1}, \tag{1.8}$$

where y_0 and y_1 represent the horizontal interfaces, Γ_0 and Γ_L, respectively. r_n and t_n are coefficients for the reflection and transmission, respectively. There are a finite number of modes to concern ourselves with. This finite collection comes from determining the real coefficients β_n in (1.3). The propagating wave modes are thus defined, and the remainder are called evanescent waves, depicting an exponential decay in the wave solution. In the previous section the problem was introduced leading to the computation of the reflection and transmission coefficients. This computation will be referred as the "forward" problem. The independent parameter is the profile, being characterized by the function $k(x_1, x_2)$, which shall be referred to from here as the **Index Profile Function**. In what follows, we will have occasion to use the

expression $\omega^2 k^2$. We define it here as $a \equiv \omega^2 k^2$, and it will be referred to as the **Squared Index Profile**. The established relation is $F(a)$, i.e,

$$F : \mathcal{A} \to \{(r_m), (t_n)\}, \tag{1.9}$$

where $m \in \Lambda_0$ for some index set for the reflections, and $n \in \Lambda_1$ for some index set for transmissions. These index sets will be defined later. After extracting the reflection and transmission coefficients the energy of the wave can be computed directly. The object is to minimize a basic least squares expression that minimizes the l_2 norm of the difference between computed reflections and the target reflections, and likewise for the target transmissions. Let r_m^* and t_n^* each represent the target reflections and transmissions of our desired structure.

The least squares cost functional to be minimized is given by

$$\min_{a \in \mathcal{A}} \sum_{\Lambda_0} \left| |r_m|^2 - |r_m^*|^2 \right|^2 + \sum_{\Lambda_1} \left| |t_m|^2 - |t_m^*|^2 \right|^2$$

The inverse problem is to find $a \in \mathcal{A}$ such that the above functional is minimal. The set

$$\mathcal{A} \equiv \{a \in L^\infty(\Omega) : a_1 \leq a(x) \leq a_2\} \tag{1.10}$$

is an admissible set of Squared Index Profile functions. There are possibly many profiles that can achieve the same minimum. This is due to the inherently large number of independent parameters compared to the output. Once discretized, the forward problem can be viewed as an under-determined system, $F : \mathbb{R}^n \to \mathbb{R}^m$, where $n \gg m$. This will be used to an advantage for deciding on optimal shape profiles.

E. Level-Sets

The fact that solutions to the above inverse problem are under-determined is an advantage, in the sense that there are possibly many minimizers for the least squares cost functional, and, hence, from this collection, certain shape profiles will be much more desirable than others. First, we note on the use of the word "shape". As a function in L^∞, we generally would not need to investigate any further, but here we are seeking more. A nano-technology lab uses methods of laser etching to create diffractive structures. Photonic crystals are built by synthetic methods. In each case, the material produced is composed of two distinct mediums. The space we are interested in is composed of distinct regions of homogeneous material, with index of refraction, k_1 or k_2 , with associated squared index of refraction, $a_1 = \omega^2 k_1^2$ and $a_2 = \omega^2 k_2^2$. This is clearly a proper subset of $\mathcal{A}$. In Dobson's work, the space to create the material structure was allowed to vary continuously from a_1 to a_2, thus a mixture was allowed. This exhibits a high degree of freedom to compute a minimal cost-functional, but is infeasible to recreate under current methods of lithography. Under the level-set approach, the admissible designs automatically are composed only of materials a_1 or a_2. Level-sets are a relatively new approach as a technique in inverse problems. The general idea is to construct a shape profile from the level-set of some surface function. Use of such techniques can be seen in the paper by Santosa [34]. The underlying ideas were developed by Sethian and Osher in the mid 80's in view of flow problems such as surface motion [37, 38, 39, 27]. Level-sets allow smooth functions to evolve and reshape according to a governing set of equations. The level-set of that function determines the index profile. Due to the dielectric medium, this yields profiles with distinct shapes. Inside/outside the shapes are the material of index a_2 and a_1, respectively. The level-set is the zero-level-set of some surface function, ϕ. It

is allowed to take on any shape limited by only the surface function. In lieu of this flexibility, it can also take on disjoint simply closed curves, encompassing different shaped regions.

For this level-set approach we implement an evolutionary algorithm, following [34]. It involves computing an update step related to the level-set function. Let $\phi_n(x_1, x_2)$ represent the nth level-set function, with the associated squared index profile function a_n. We consider an update step in the level-set function,

$$\phi_{n+1} = \phi_n + \delta\phi,$$

which produces an associated update in a_n,

$$a_{n+1} = a_n + \delta a.$$

The governing equations are those that guarantee a decrease in the cost functional of the forward problem. We describe this method more thoroughly in Chapter VI.

CHAPTER II

COMPUTATION OF THE HELMHOLTZ EQUATION

In this chapter we discuss methods for computing the solution to the modified Helmholtz equation. We define the multi-layered structures and the numerical schemes used in the approximation. For the purposes of this chapter we consider the full L-layered domain Ω, as defined from the previous chapter, with top boundary Γ_0 and bottom boundary Γ_L. The modified Helmholtz problem can be stated as follows: Find u in $H^1(\Omega)$ that solves the following PDE and BC's,

$$\Delta_\alpha u + k^2\omega^2 u = 0, \qquad \text{in } \Omega, \tag{2.1}$$

$$\frac{\partial u}{\partial x_2} - T_0 u = -2i\beta_0, \quad \text{on } \Gamma_0, \tag{2.2}$$

$$\frac{\partial u}{\partial x_2} + T_L u = 0, \qquad \text{on } \Gamma_L. \tag{2.3}$$

The expression on the right in (2.2) is $T_0(f)$, where $f(x)$ is the incoming plane wave on the top boundary. The approach taken here is to approximate the solution to equations (2.1) - (2.3) by a standard finite element discretization using a rectangular mesh. For simplicity we build a uniform rectangular mesh. In more advanced finite element schemes, unstructured triangular meshes are designed (eg. see [36, 8] for details). For ease of readability, we consider

$$k \equiv k(x_1, x_2). \tag{2.4}$$

We view this as an element in $L^\infty(\Omega)$. We need not assume any kind of regularity on the Index Profile for existence of a solution. This will be presented in the next theorems. We assume only that $a_1 \leq \omega^2 k^2 \leq a_2$. It is understood that k is constant along the boundaries, Γ_0 and Γ_L.

Next, notice the use of the Dirichlet-Neumann (D-N) operators, T_j, $j \in \{0, L\}$. These operators are used to guarantee that only outgoing waves (except for the single incoming source term) are present. Another way to state this is that there are no scattered waves coming in from infinity. This formulation replaces the familiar Sommerfeld radiation condition dictating how waves behave at infinity. Specifically, for 2-D the Sommerfeld condition is [6]

$$\lim_{r\to\infty} \sqrt{r}(\frac{\partial u}{\partial r} - iku) = 0.$$

With the D-N operators, the wave behavior at infinity is prescribed as a boundary condition on the transparent interfaces. We define the following notation:

Definition 2.1

Let Λ_0 denote the finite set of n's that yield the reflection modes, that is,

$$\Lambda_0 = \{n : \omega^2 k^2 - (n+\alpha)^2 > 0\},$$

and likewise, let Λ_1 denote the finite set of n's that yield transmission modes.

Note: $\Lambda_0 = \Lambda_1$ for $k|_{\Gamma_0} = k|_{\Gamma_L}$. Note also that Λ_0, Λ_1 are always nonempty due to the definition of α. We now present the variational form and discuss its finite element discretization.

A. Variational Form

The Helmholtz equation and BC's, (2.1)-(2.3) has the following standard variational form:

Problem 2.2 (Variational Equation)

Find $u \in H^1(\Omega)$ such that for every $v \in H^1(\Omega)$ we have

(2.5)

$$\int_\Omega \nabla u \cdot \overline{\nabla v} - \int_{\Gamma_0} T_0 u\overline{v} - \int_{\Gamma_L} T_L u\overline{v} - 2i\alpha \int_\Omega \tfrac{\partial}{\partial x_1} u\overline{v} - \int_\Omega (k^2\omega^2 - |\alpha|^2) u\overline{v} = -2i\beta_0 e^{-i\beta_0^0 y_0} \int_{\Gamma_L} \overline{v}.$$

Using finite element methods, we discretize the domain into a uniform rectangular grid. The nodal basis functions are bilinear real valued functions $\phi(x)\psi(y)$. Due to the periodicity of the domain, there are no degrees of freedom on the right hand boundary. We define the rectangular grid to be an $M \times N$ set of rectangles. The nodal space is set at the corner points and this will be $(M+1) \times N$. For reasons to be addressed in Chapter VII, we consider k to be piecewise constant on each rectangular grid where the index of refraction is either k_1 or k_2, but is split along the level-set. How it is split will be addressed in Chapter VII.

As a multi-layered structure, the interfaces, Γ_j, where $0 \leq j \leq L$, define the boundaries between adjacent layers. As light passes through each interface, we have a characterization of the wave profile via the modes of propagation.

Definition 2.3 (Modes of Propagation)

For $n \in \mathbb{N}$, let

$$\beta_n^j = \sqrt{\omega^2 k^2 - (n+\alpha)^2},$$

where $j = \{0, 1, \ldots, L\}$. The value of k is determined by the particular layer interface, i.e., $k|_{x_2=-jd}$. When $\omega^2 k^2 - (n+\alpha)^2 < 0$, we let $\beta_n^j = i\sqrt{|\omega^2 k^2 - (n+\alpha)^2|}$.

Associated with these modes are corresponding D-N operators. The Dirichlet-Neumann operators are defined as follows:

Definition 2.4 (Dirichlet-Neumann Operators (D-N))

$$(T_j u)(x_1) = \sum_{n \in Z} i\beta_n^j u_n e^{inx_1},\ j = 0, \ldots, L,$$

where u_n are the Fourier coefficients of u, computed along the interfaces Γ_j.

Lemma 2.5

The operators T_j are linear and continuous from $H^{\frac{1}{2}}(\Gamma_j) \to H^{-\frac{1}{2}}(\Gamma_j)$.

Proof. Linearity is trivial. Continuity can be seen from the definition of Sobolev spaces and the fact that $|\beta_n| \sim |n|$. □

From Definition 2.4 notice that the T_j operators are non-local. Of course, in the corresponding finite element matrices this results in dense sub-matrices at the boundaries.

In [10] existence and uniqueness of solutions to (2.1)-(2.3) were proved. However, a slight modification of the proof will be presented here. We state the result for completeness.

Theorem 2.6 (Existence and Uniqueness)

Suppose that $k \in L^\infty(\Omega)$ with $k_1 \leq k(x) \leq k_2$. Then there exists a ω_0, such that for all $\omega < \omega_0$, there exists a unique weak solution $U_\alpha \in H^2(\Omega)$ for the variational problem (2.5). Moreover, $\|U_\alpha\|_{H^2(\Omega)}$ is bounded independent of k.

The theorem says that a bounded invertible operator exists, and this bound is uniform over all profiles k, provided the frequencies are sufficiently small.

During the course of the proof, all constants, C, are understood to change. Also, all norms, $\|\cdot\|$, without subscripts are understood to be standard $L^2(\Omega)$-innerproduct norms. Essential in the proof of Theorem 2.6 is the following:

Let $B_1(u,v)$ denote the sesquilinear form,

$$\int_\Omega \nabla u \cdot \overline{\nabla v} - \int_{\Gamma_0} T_0 u\overline{v} - \int_{\Gamma_L} T_L u\overline{v} - 2i\alpha \int_\Omega (\frac{\partial}{\partial x_1}u)\overline{v} + \int_\Omega (|\alpha|^2)u\overline{v}. \tag{2.6}$$

Theorem 2.7

For $\omega < \omega_0$, for some sufficiently small ω_0, $B_1(u,v)$, where $u, v \in H^1(\Omega)$, is bounded and coercive.

Proof. We first prove continuity. By the Trace Theorem [2] and continuity of the D-N operators and several applications of the Schwartz inequality we get

$$|B_1(u,v)| \leq \|\nabla u\|\|\nabla v\| + c_1\|u\|\|v\| + c_2\|\nabla u\|\|v\| \leq C\|u\|_{H^1(\Omega)}\|v\|_{H^1(\Omega)}$$

Next, let us consider $u \in C^1(\Omega)$. Let $\bar{u}(y) = \frac{1}{2\pi}\int_0^{2\pi} u(s,y)ds$. By continuity, there exists $a \equiv a(y) \in [0, 2\pi]$ such that $u(a,y) = \bar{u}(y)$. Then,

$$\bar{u}(y) - \bar{u}(0) = \int_0^y \bar{u_y}(t)dt = \int_0^y \frac{1}{2\pi}\int_0^{2\pi} u_y(s,t)ds\,dt.$$

Hence,

$$|\bar{u}(y) - \bar{u}(0)|^2 \leq \int_0^{2\pi} \frac{1}{2\pi}\int_0^{2\pi} |u_y(s,t)|^2 ds\,dt \leq C\|\nabla u\|^2,$$

and

$$\|\bar{u}(y) - \bar{u}(0)\|^2 = \int_0^{2\pi} |\bar{u}(y) - \bar{u}(0)|^2 dy \leq C\|\nabla u\|^2. \tag{2.7}$$

By the Fundamental Theorem of Calculus,

$$u(x,y) - u(a,y) = \int_a^x u_x(t,y)dt,$$

hence,

$$|u(x,y) - u(a,y)|^2 \leq \int_a^x |u_x(t,y)|^2 dt \leq \int_0^{2\pi} |u_x(t,y)|^2 dt.$$

Finally,

$$\int_0^{2\pi} |u(x,y) - \bar{u}(y)|^2 dy \leq \int_0^{2\pi} \int_0^{2\pi} |u_x(t,y)|^2 dt\, dy \leq \|\nabla u\|^2,$$

so that

$$\|u(x,y) - \bar{u}(y)\|^2 \leq C\|\nabla u\|^2. \tag{2.8}$$

By combining expressions (2.7) and (2.8), the triangle inequality yields

$$\|u - \bar{u}(0)\|_2 \leq C\|\nabla u\|_2.$$

And, hence, $\|u\|_2 \leq C\|\nabla u\|_2 + \|\bar{u}(0)\|_2$. Since $c_1^2|u_0|^2 = \|\bar{u}(0)\|^2$, it follows that

$$\|u\|_2 \leq c_1|u_0| + C\|\nabla u\|_2. \tag{2.9}$$

Next, we look at the imaginary part of $B_1(u,u)$, $-\sum_n \beta_n |u_n|^2$. Thus,

$$-Im(B_1(u,u)) \geq \beta_0|u_0|^2 = c\omega|u_0|^2.$$

Now, $\|u\|_{H^1}^2 \leq 2(c_1^2|u_0|^2 + c_2^2\|\nabla u\|^2)$. From this, and using the bound on the imaginary part of B_1, and if we let ω_0 be fixed and positive, then we get for $\omega \leq \omega_0$,

$$\begin{aligned} c\omega\|u\|_{H^1}^2 &\leq C\omega(|u_0|^2 + \|\nabla u\|^2) \\ &\leq |Im(B_1(u,u))| + C\omega_0|Re(B_1(u,u))| \leq C_1|B_1(u,u)| \end{aligned} \tag{2.10}$$

The function u is understood to be in $C^1(\Omega)$. By letting $u \in H^1$, and letting $u_n \in C^1$, for each n, such that $u_n \to u$, it follows from the density of C^1 in H^1 and

standard convergence arguments that $B_1(u,u) \geq C\|u\|^2_{H^1}$. $\square$

The above argument utilized the dependence on the frequency, ω. Now, we prove the existence and uniqueness for a wave solution.

Proof of Theorem 2.6

Proof. Recalling the variational form, (2.5), we break it apart into two sesquilinear forms, $B_1(u,v)$ as described in (2.6), and

$$B_2(u,v) = \int_\Omega k^2 u\overline{v}.$$

(Note, ω^2 is omitted.)

Consider the variational equation,

$$B_1(u,v) = f(v), \tag{2.11}$$

where f is a bounded linear operator on $H^1(\Omega)$. From the Riesz Representation Theorem [33] there exists a bounded linear map, $A_1 : H^1 \to H^{-1}$, such that $\langle A_1 u, v\rangle_{H^1} = B_1(u,v)$, for all $v \in H^1$. The mapping A_1 can be viewed as a dual space pairing between $H^1(\Omega)$ and $H^{-1}(\Omega)$. Further, we have from Riesz that there exists $z \in H^{-1}$ such that $\langle z, v\rangle = f(v)$. Since we are seeking a solution u such that (2.11) holds for all v, this is equivalent to the operator equation $A_1 u = z$. From the Lax-Milgram Theorem we obtain that there does exist a unique solution $u \in H^1$, i.e., the operator A_1 is invertible. and $u = A^{-1}z$. Further, this inverse is bounded as follows. Since $|B_1(u,u)| \geq c\omega\|u\|^2_{H^1}$,

$$c\omega\|u\|^2_{H^1} \leq |\langle A_1 u, u\rangle| \leq \|A_1 u\| \cdot \|u\|,$$

so we get

$$c\omega\|A^{-1}z\| \leq \|z\|.$$

By taking the supremum over all $\|z\| = 1$ on the last expression, it follows that $\|A^{-1}\| \leq \dfrac{1}{c\omega}$.

Next, we consider the sesquilinear form, $B_2(u, v)$. Since $k_2 \geq k \geq k_1$, B_2 is clearly bounded. Further, this bound is independent of the particular $k \in L^\infty(\Omega)$, due to the above bounds on k. Hence, by similar reasoning for A_1, there is a bounded operator, $A_2 : H^1 \to H^{-1}$. We denote this bound on A_2 by a constant $\hat{C}$. Further, this operator is compact, due to Rellich's Lemma and the definition of dual space norms. Now, we consider the operator $A = A_1 - \omega^2 A_2$. Since A_1 is invertible, we factor it out, and we have $A_1(I - \omega^2 A_1^{-1}A_2)$. The resulting operator, $(I - \omega^2 A_1^{-1}A_2)$, is invertible provided $\|\omega^2 A_1^{-1}A_2\| < 1$. Since $\|A_1^{-1}\| \leq \dfrac{1}{c\omega}$, and A_2 is bounded, it follows that for $\omega_0 > 0$ chosen sufficiently small, the A operator is invertible for all $\omega \leq \omega_0$. Further, the bound on the operator A^{-1} can be seen as follows:

$$\|A^{-1}\| \leq \frac{1}{c\omega(1 - \frac{\omega\hat{C}}{c})}. \tag{2.12}$$

This bound is uniform over all profiles $k \in \mathcal{A}$, and for $\omega \leq \omega_0$. Additionally, the choice of the value ω_0 depended only on the bound on the profiles k, and not on any specific k.

So far, we have established the solution u lies in $H^1(\Omega)$. However, we have more here. From a distributional sense, the Helmholtz equation, $\Delta_\alpha u + k^2 u = 0$, gives that $\Delta_\alpha u$ shares the same regularity as $-k^2 u$, which is guaranteed to be in $L^2(\Omega)$. Thus, $u \in H^2(\Omega)$. □

We describe the error of the discrete solution. First, we begin by describing some standard initial properties for a finite element space: We wish to find a solution $u_\alpha \in H^1(\Omega)$ such that

$$a(u_\alpha, \phi) = (f, \phi), \quad \forall \phi \in H^1(\Omega); \tag{2.13}$$

The sesquilinear form, $a(\cdot, \cdot)$, is defined by equation (2.5). The function f is in $H^{-1}(\Omega)$, and $(\cdot, \cdot)$ represents dual pairing. The function k, defined above (2.4), is in $L^\infty(\Omega)$.

Let $\{S^h : h \in (0, 1]\}$ denote a family of finite-dimensional subspaces of $H^1(\Omega)$, where h represents the maximum mesh size. Define the semi-norm $|\cdot|_m$, m a non-negative integer, as $|\cdot|_m = \max_{|s|=m} \|D^s \cdot\|_{L^2(\Omega)}$. The finite dimensional variational equivalent of (2.13) (Ritz-Galerkin [8]) is to find $u^h \in S^h$, such that for all $v^h \in S^h$,

$$a(u^h, v^h) = (f, v^h).$$

Define $e^h = u - u^h$. Let us assume the following for S^h: For $\nu \in H^l(\Omega), l \geq 2$,

$$\begin{aligned} \inf_{\Psi \in S^h} \quad & (\|\nu - \psi\|_{L^2(\Omega)} + h|\nu - \psi|_1 + h^{\frac{1}{2}}\|\nu - \psi\|_{L^2(\Gamma_1)} + h^{\frac{1}{2}}\|\nu - \psi\|_{L^2(\Gamma_2)} \\ & + h\|\nu - \psi\|_{H^{\frac{1}{2}}(\Gamma_1)} + h\|\nu - \psi\|_{H^{\frac{1}{2}}(\Gamma_2)}) \leq Ch^{l'}\|\nu\|_{H^{l'}(\Omega)}, \end{aligned} \tag{2.14}$$

where C is a constant independent of h and ν, and l' is any integer in $[2, l]$. This establishes an approximation assumption on the type of finite element spaces we are working with. Specifically, it gives that our finite space of bilinears along with the uniform mesh is an adequate space for approximation as defined in the above expression. For details, see [4]. Here, we present the main result on the finite element approximations. For proof, see [4].

Theorem 2.8

Suppose that $u \in H^l(\Omega)\,(l \geq 2)$ satisfies the above variational form with $k \in L^\infty$.

Suppose also that the family of finite element spaces S^h satisfies the assumption (2.14), Then there exists $h_0 \in (0,1]$ such that for $h \in (0,h_0)$ the variational form (2.5) admits a unique solution, u^h. Moreover, the following estimates hold:

$$\begin{aligned} \|e^h\|_{L^2(\Omega)} &\leq Ch^l \|u\|_{H^l(\Omega)}, \\ \|e^h\|_{H^1(\Omega)} &\leq Ch^{l-1} \|u\|_{H^l(\Omega)}, \end{aligned}$$

where C depends on $\|k\|_{L^\infty}$ but is independent of h and u.

B. Truncated Dirichlet-Neumann Operators

In the finite element approximations, a further approximation is made on the problem. The Dirichlet-Neumann (D-N) operators are truncated at a specified $\hat{N}$. Recalling Definition 2.4, we define the new operators as

Definition 2.9 (Truncated D-N Operators)

$$(T_j^{\hat{N}} u)(x_1) = \sum_{|n| \leq \hat{N}} i\beta_n u_n e^{inx_1},\ j = 0, L.$$

The question of existence and uniqueness of solution with the truncated operators has been addressed in [4]. The answer is affirmative in both cases, for $\hat{N}$ sufficiently large. Letting $a_{\hat{N}}(u,\phi)$ represent the sesquilinear form analog to $a(u,\phi)$ (2.13) with truncated D-N operators, we seek solutions to

$$a_{\hat{N}}(u,\phi) = f(\phi), \quad \forall \phi \in H^1. \tag{2.15}$$

Recall from (2.1) the definition of Λ_j. We let Λ_j^- be the set of indices $\{|n| \leq \hat{N}\} \bigcap \widetilde{\Lambda_j}$. I.e., the set of indices less than $\hat{N}$ and with β_n^j purely imaginary. Also, recall $\Gamma_0 = \{(x_1,x_2) : x_2 = y_0\}$, and $\Gamma_1 = \{(x_1,x_2) : x_2 = y_1\}$. Then let $\Gamma_0' = \{(x_1,x_2) : x_2 = y_0 - b\}$, where $b > 0$, and $\Gamma_1' = \{(x_1,x_2) : x_2 = y_1 + b\}$. Also, we define $u_{\hat{N}}^h$ to be

the finite element solution with the truncated boundary operators to problem (2.15). Finally, let $e^h_{\hat{N}}$ denote the error in the truncated solution, $u - u^h_{\hat{N}}$.

Theorem 2.10 (Bao [4])

Suppose that $u \in H^l(\Omega)\,(l \geq 2)$ satisfies (2.13) with $k \in L^\infty(\Omega)$. Suppose also that the family of finite element spaces S^h satisfies (2.14). Then there exist $h_0 \in (0, 1]$ and an integer N_0, such that for $h \in (0, h_0)$ and $\hat{N} \geq N_0$, problem (2.13) attains a unique solution $u^h_{\hat{N}}$. Moreover, the following estimates hold:

$$\|e^h_{\hat{N}}\|_{L^2} \leq Ch \sum_{j=0,1} e^{-b\sqrt{(\hat{N}-|\alpha|)^2-k_j^2}} \|u\|_{H^{\frac{1}{2}}(\Gamma'_j)} + (C(k)h + C\hat{N}^{-1/2})\|e^h\|_{H^1(\Omega)} \tag{2.16}$$

and

$$\|e^h_{\hat{N}}\|_{H^l(\Omega)} + \left(\sum_{n \in \Lambda_j^-} (-i\beta_n^j) e^h_{\hat{N}n}|_{\Gamma_j} \right)^{1/2} \leq C(k)h^{l-1}\|u\|_{H^l(\Omega)} + C \sum_{j=0,1} e^{-1/2(b)\sqrt{(\hat{N}-|\alpha|)^2-k_j^2}} \|u\|_{H^{1/2}(\Gamma'_j)}, \tag{2.17}$$

where C depends on $\|k\|_{L^\infty}$, but is independent of h, $\hat{N}$, and u.

From here on, the notation on the truncated D-N operators will drop the subscript $\hat{N}$. We will assume the D-N operators are truncated in the finite case and are not truncated in the continuous case as in (2.13).

C. Multi-Layered Solutions

The problem (2.1)-(2.3) has been formulated as a multi-layered structure with the single domain $\Omega = \bigcup_{i=0}^{L-1} \Omega_i$, What we will be ultimately concerned with is realization of the parameter space. Specifically, the profile space is viewed as a repeated layer in Ω, As a function, $a \in \mathcal{A}$, it will vary only within Ω_0, but still reside in $L^\infty(\Omega)$ after extending it for the L layers. In Chapter IV, the profile space will be modified to

allow for a repeated parameter space, and the problem will be reformulated.

D. Size of Mesh

Next, we discuss some computational issues. While it follows from Theorem 2.8 that the L^2 error of the solution is improved with h^2, where h is the mesh size, it is not necessary to make it increasingly small. In practice, it is generally understood that one should choose $h < \lambda/5$, where λ is the wavelength within the medium. The reason for this is that to preserve a wave's profile, at least 5 points will model the wave along a single wavelength. It is clear that the size of the problem (number of unknowns) grows as $\frac{1}{h^2}$. Hence, in our computations, we will consider moderately sized problems on the order of 16×16, 32×32, 64×64. Indeed, as the iterations increase in the optimization, time constraints prohibit the finer meshes.

CHAPTER III

THE INVERSE PROBLEM AND OPTIMAL DESIGN

The inverse problem and the forward problem are closely related with each other. The forward problem is usually characterized as computing directly the solution of a model equation given certain inputs to the problem, i.e., computing the wave solution or the image given an incoming wave and a scattering source. Inverse problems, in general, are the reverse. The inverse problem is to determine parameters that are responsible for the measured data. For the above two examples, given the blurred image, or the wave solution (like the far-field behavior on a background screen), we compute what the scattering source looks like, or what the original image is. Typically, there are many parameters that determine the outcome in the forward problem. The inverse problem, by contrast, is concerned with certain specific parameters that affect the outcome, while leaving the others fixed. For example, a unit source wave or its angle of incidence (or both) may be kept fixed while trying to determine the specific scattering profile. The point here is that there may be different inverse problems for a given forward problem, depending on the question being asked.

The forward problem in this thesis is the following: Given an incoming wave f, and a squared index profile a, compute the reflected and transmitted modes. The inverse problem then is to construct a layered interface profile a that produces a desired distribution of reflections and transmissions (the scattering), denoted r_n^* and t_n^*, respectively, with a fixed incoming wave. The primary focus of this thesis is to analyze and test the feasibility of the method used to reconstruct a. But reconstruction is not the appropriate word to use in this paper, for this suggests that a unique scattering profile exists. From an engineering perspective, the word **synthesis** is sometimes used. In the sense of optimal design, it is often desirable to have many solutions from

which to choose the one more appropriate for a given situation, i.e, smaller shapes or a smoother boundary. It is the aim of this chapter to demonstrate the ill-posedness of the problem and how this benefits the optimal shape strategy. To affect the optimization, a cost function is used to measure the profile's scattering properties. In the spirit of [34] and [10], we will generate a least squares cost functional for the coefficients of reflection and transmission.

A. The Reflection and Transmission Coefficients

First, let us consider the nature of the propagating reflection and transmitted waves. As will be shown, there are a finite number of such waves while the remainder are evanescent (exponentially decaying). The distribution of energy can be altered through the choice of the squared index profile function a. Fortunately, the number and direction of each propagating wave is known independent of the particular profile, a. This appears somewhat surprising, but as will be seen, is a consequence of Floquet theory. That is, the periodicity of the solutions dictate the nature of the reflection and transmitted waves.

As described in the introduction, each layer is separate from the next via a boundary interface, denoted Γ_j. This boundary is in a homogeneous region of constant index profile. On the interfaces we analyze the solution being a sum of analytic waves of the form $e^{\pm i(\beta_n x_2 + a_n x_1)}$.

The outgoing modes for the reflections and transmission waves are realized through the β_n and a_n coefficients. This is a matter of the 2-D vector $\langle a_n, \beta_n \rangle$ being the normal component of the plane wave front. It will be readily seen that the coefficients a_n are integers.

Consider the solution $U_\alpha(x_1, x_2)$. Since it is periodic in x_1 let it have Fourier de-

composition $U_\alpha(x_1, x_2) = \sum_n u_n(x_2)e^{inx_1}$. Substitution into the modified Helmholtz equation yields the following:

$$[u_n''(x_2) + \beta_n^2 u_n(x_2)]e^{inx_1} = 0,$$

where $\beta_n^2 = k^2 - (n+\alpha)^2$. In the homogeneous region around the boundaries, Γ_0, Γ_1, we conclude that $u_n(x_2) = c_n e^{\pm i\beta_n x_2}$, where c_n is a real coefficient.

For all but finitely many values of n, β_n is a pure imaginary number. Exponential decay results for the appropriate sign, depending on whether we consider a reflection or a transmission. I.e., for reflections, we consider only $u_n(x_2) = r_n e^{+i\beta_n x_2}$, for as a wave travels upward, x_2 will increase. These solutions are known as evanescent waves. For the finite remainder of n's, we have determined the possible modes for the reflection and transmission waves. Hence, the solutions are a sum of a finite number of propagating modes with an infinite sum of decaying evanescent waves.

Recall from Chapter II the indices for propagation modes (2.1), Λ_j. Often is the case that the transmission and reflection media are the same (air, for example with index $k_1 = 1$). This would yield that $\Lambda_0 = \Lambda_1$.

Finally, the solutions were considered in a neighborhood of the boundaries. Above Γ_0 and below Γ_L, the homogeneous regions extend indefinitely, thus, uniqueness of the analytic solutions requires these solutions extend indefinitely as well.

The reflectance and transmission coefficients for the solution, U_α are computed as follows: The solution at the top and bottom boundary read as (see (1.8))

$$U_\alpha \mid_{\Gamma_0} = \sum_{n\in\mathbb{Z}} r_n e^{inx + i\beta_n^0 y_0} + e^{-i\beta^0 y_0}, \tag{3.1}$$

$$U_\alpha \mid_{\Gamma_1} = \sum_{n\in\mathbb{Z}} t_n e^{inx - i\beta_n^1 y_1}, \tag{3.2}$$

where y_0 and y_1 represent the horizontal interfaces as described earlier. Extracting

the coefficients $(r_n,\, t_n)$ amounts to computing the Fourier coefficient of the solution at the respective boundary:

$$
\begin{aligned}
(3.3) \qquad r_n &= \frac{e^{-i\beta_n^0 y_0}}{2\pi}\int_{\Gamma_0}(U\mid_{\Gamma_0})e^{-inx_1}dx + \begin{cases} -e^{-2i\beta^0 y_0},\ n=0 \\ 0,\ otherwise, \end{cases} \\
t_n &= \frac{e^{+i\beta_n^1 y_1}}{2\pi}\int_{\Gamma_L}(U\mid_{\Gamma_1})e^{-inx_1}dx.
\end{aligned}
$$

B. Conservation of Energy

In correctly formulating the problem and keeping track of valid solutions, we monitor the conservation properties of the system. In our problem, we have loss-less dielectric material. As light passes through the structure, there should be no energy loss. Thus the energy distribution of the system is balanced between the reflection and transmission coefficients. The conservation of energy of the system can be characterized by the following formula. The interesting feature about this formula is that it is exact for the finite element solutions as well, $U^h \in S^h$. First, we define the profile of the incoming wave along the top interface as f, and likewise, g, for the incoming wave along the bottom interface. Further, let f_n, and g_n, denote the Fourier coefficients of f, and g, respectively.

Theorem 3.1 (Conservation of Energy)

Let f_n and g_n be defined as above, and let r_n, t_n be defined as in (3.3). Then,

$$
(3.4) \qquad \sum_{n\in\Lambda_0}\beta_n^0\left|r_n\right|^2 + \sum_{n\in\Lambda_1}\beta_n^1\left|t_n\right|^2 = \sum_{n\in\Lambda_0}\beta_n^0\left|f_n\right|^2 + \sum_{n\in\Lambda_1}\beta_n^1\left|g_n\right|^2.
$$

This reduces to the familiar conservation of energy, when $f = 1$ and $g = 0$, which corresponds to unit energy of incoming wave incident from above and no wave from

below. In this case, divide by $\beta_0 \equiv \beta_0^0$. Thus,

$$\sum_{n\in\Lambda_0} \frac{\beta_n^0}{\beta_0} |r_n|^2 + \sum_{n\in\Lambda_1} \frac{\beta_n^1}{\beta_0} |t_n|^2 = 1 \tag{3.5}$$

We prove (3.4):

Proof. From the variational form,

$$\int_\Omega \nabla u \cdot \overline{\nabla v} - 2i\alpha \int_\Omega \partial_1 u\overline{v} - \int_\Omega (k(x_1,x_2)^2 - \alpha^2) u\overline{v}$$
$$- \int_{\Gamma_0} Tu\overline{v} - \int_{\Gamma_1} Tu\overline{v} = -2\int_{\Gamma_0} Tf\overline{v} - 2\int_{\Gamma_1} Tg\overline{v},$$

we set v to u, and we look for the imaginary part of the expression. This essentially eliminates the top line in the equation above. Breaking apart the T-operator, we have the following:

$$-\int_{\Gamma_0}\sum_{\Lambda_0} i\beta_n^0 u_n e^{inx}\overline{u} - \int_{\Gamma_1}\sum_{\Lambda_1} i\beta_n^1 u_n e^{inx}\overline{u} = -2\int_{\Gamma_0}\sum_{\Lambda_0} i\beta_n^0 f_n e^{inx}\overline{u} - 2\int_{\Gamma_1}\sum_{\Lambda_1} i\beta_n^1 g_n e^{inx}\overline{u}.$$

Simplifying, this becomes

$$\int_{\Gamma_0}\sum_{\Lambda_0} \beta_n^0 u_n e^{inx}\overline{u} + \int_{\Gamma_1}\sum_{\Lambda_1} \beta_n^1 u_n e^{inx}\overline{u} = +2\int_{\Gamma_0}\sum_{\Lambda_0} \beta_n^0 f_n e^{inx}\overline{u} + 2\int_{\Gamma_1}\sum_{\Lambda_1} \beta_n^1 g_n e^{inx}\overline{u}. \tag{3.6}$$

Notice that we can distribute the integral into the summations. Factoring out of the integral all constants, we arrive at

$$\sum_{n\in\Lambda_0} \beta_n^0 u_n \int_{\Gamma_0} e^{inx}\overline{u} + \sum_{n\in\Lambda_1} \beta_n^1 u_n \int_{\Gamma_1} e^{inx}\overline{u} = 2\sum_{n\in\Lambda_0} \beta_n^0 f_n \int_{\Gamma_0} e^{inx}\overline{u} + 2\sum_{n\in\Lambda_1} \beta_n^1 g_n \int_{\Gamma_1} e^{inx}\overline{u}.$$

Notice further that the integrals are now the conjugate Fourier coefficients of the function along the boundary (modulo a factor of 2π which is omitted below)

$$\sum_{n\in\Lambda_0} \beta_n^0 |u_n|^2 + \sum_{n\in\Lambda_1} \beta_n^1 |u_n|^2 = 2\sum_{n\in\Lambda_0} \beta_n^0 f_n \overline{u_n} + 2\sum_{n\in\Lambda_1} \beta_n^1 g_n \overline{u_n}. \tag{3.7}$$

After factoring i out of the expression we are looking for the real values from above. The function u should consist of incoming plus outgoing waves. On the boundary Γ_0, this means we look for $u(x) = \sum (f_n + r_n)e^{inx} = \sum u_n e^{inx}$. On the boundary Γ_L we have $u(x) = \sum (g_n + t_n)e^{inx} = \sum u_n e^{inx}$. Upon substitution into equation (3.7), we will subtract one sum from each boundary from the right hand side to get the following:

(3.8)

$$\sum_{n\in\Lambda_0} \beta_n^0(|r_n|^2 + r_n\overline{f_n} - f_n\overline{r_n}) + \sum_{n\in\Lambda_1} \beta_n^1(|t_n|^2 + g_n\overline{t_n} - t_n\overline{g_n}) = \sum_{n\in\Lambda_0} \beta_n^0|f_n|^2 + \sum_{n\in\Lambda_1} \beta_n^1|g_n|^2.$$

Taking only the real values yields equation (3.4). □

In the finite element approximations, the above arguments follow through exactly except how the T operators are split apart. In the finite element case, the operators are truncated at a finite $\hat{N}$, see (2.9). Since only the imaginary parts are kept in the above proof, it suffices to let $\hat{N}$ be large enough to account for Λ_0 and Λ_1.

C. Least Squares Functional

We now define the least squares functional, $J(a)$. As was given in Chapter I, it is a straight forward function. First, recall r_m^*, $m \in \Lambda_0$, and t_m^*, $m \in \Lambda_1$ are the desired reflections and transmission energy modes. We will refer to these as the target modes.

Next we define the forward map F that computes the outgoing reflection and transmission modes. This was briefly defined in Chapter I, (1.9). We reintroduce it here and make a slight modification. Recall the propagation modes for the scattered wave solution (3.3). From the section on conservation of energy, the modes were

transformed into the energy modes by appropriate scaling [1] (3.5). Namely,

$$r_n \to \frac{\beta_n^j}{\beta_0}|r_n|^2.$$

We utilize this map to define the function

$$F : \mathcal{A} \subset L^2(\Omega) \to \mathbb{R}^m, \tag{3.9}$$

where $m = |\Lambda_0| + |\Lambda_1|$, by $F_n = \frac{\beta_n^j}{\beta_0}|r_n|^2$. The function $F(a)$ then represents the vector valued components of the propagating modes from the profile, $a \equiv \omega^2 k^2$, the squared refractive index. We will refer to this function as the forward map. The associated least squares cost functional is given by

$$J(a) = \sum_{\Lambda_0} \left| \frac{\beta_m^0}{\beta^0} |r_m|^2 - |r_m^*|^2 \right|^2 + \sum_{\Lambda_1} \left| \frac{\beta_m^1}{\beta^0} |t_m|^2 - |t_m^*|^2 \right|^2. \tag{3.10}$$

Thus, $J(a)$ measures the total difference (in the least squares sense) between the target modes and the generated reflection and transmitted modes from the profile a. Notice the explicit dependence on the index profile a. The right hand side implicitly depends on the profile. As explained previously, the modes (i.e., the sets Λ_0 and Λ_1) do not depend on the index, thus the dependency is well defined through the reflection and transmission parameters.

Since the conservation of energy (3.5) guarantees that the reflection and transmission coefficients will sum to unity, provided they are scaled by the outgoing modes β_n^j, this allows one to view the target modes in the space $\{x \in \mathbb{R}^m : \sum x_j = 1\}$. Thus, we let $\mathbf{q} \equiv [|r_n^*|^2, |t_n^*|^2]$. Using F directly, the cost function is equivalently

$$J(a) = \|F(a) - \mathbf{q}\|^2.$$

[1] We are more concerned with the energy of a given mode rather than the specific reflection/transmission coefficient, since phase information is difficult to measure at optical wavelengths.

The least squares functional J is well defined for the solution U_α was shown to be unique in Chapter II. As will be seen, F also has sufficient smoothness to establish a gradient. As with any derivative-based method this is an essential feature for the level-set method.

D. Stability of the Forward Problem

In this section we study the sensitivity of the forward problem. Its purpose is two fold in that (1) we would like to provide an analysis of the perturbation of the level-set and (2) how it can be controlled to affect certain changes in the propagation modes. First, we discuss the continuity of the problem. The following theorem is a consequence of the Frechét differentiability of the forward map, F. The proof will be deferred until Chapter IV. First, recall the coefficients, a_1 and a_2 from the Squared Index Profile (1.10).

Theorem 3.2 (Continuity of the Helmholtz solution)
Let ω be a given frequency for the incoming wave. Let $U[a]$ denote the $H^1(\Omega)$ solution to the variational problem (2.5), with explicit dependence on the profile a such that $a_1 \leq a \leq a_2$. Let $\delta a \in L^\infty$, such that $a_1 \leq a + \delta a \leq a_2$. Then the following estimate holds:

$$\|U[a+\delta a] - U[a]\|_{H^1} \leq C\|\delta a\|_\infty. \tag{3.11}$$

The constant C is independent of the profile a, provided a sufficiently small ω_0 is given as in Theorem 2.6, and $\omega \leq \omega_0$.

From this estimate, and the uniform boundedness of the operator as described in Chapter II, it can be shown that the reflection modes are Weak-$^*L^\infty(\Omega)$ continuous (see [10] for details). In the next theorem, continuity is expressed in terms of a bound.

Theorem 3.3 (Continuity of the Forward map)

Let $\hat{r}_n$ represent the reflection mode, $r_n[a+\delta a]$, and let $r_n = r_n[a]$. Then for each n,

$$|\hat{r}_n - r_n|^2 \leq C\|\delta a\|_\infty^2.$$

Proof. From Parseval's equality and the completeness of the periodic Fourier functions, e^{inx}, we have

$$|\hat{r}_n - r_n|^2 \leq \sum_{n\in\mathbb{Z}} |\hat{r}_n - r_n|^2 \leq C\int_\Gamma |\hat{U} - U|^2.$$

From Sobolev inequality on traces and the Trace theorem (see [2]), we have

$$\int_\Gamma |\hat{U} - U|^2 \leq C\|\hat{U} - U\|^2_{H^{\frac{1}{2}}(\Gamma)} \leq C\|\hat{U} - U\|^2_{H^1(\Omega)} \leq C\|\delta a\|^2.$$

The last inequality follows from the continuity of U, Theorem 3.2. □

E. Ill-Posed and Optimal Shapes

What we establish in this section is the feasibility of searching for a solution a in the domain space $\mathcal{A} \subset L^\infty(\Omega)$.

First, from Chapter II, we established the uniqueness of the solution $U_\alpha \in H^2(\Omega)$. The question of stability of the solution on the profile will be covered here. We analyze how the domain space may allow for multiple solutions. This ill-posedness in the inverse problem is an advantage in optimal design. Within the framework of a system of equations, it is clear that an under-determined system exists with the profile space being the set of independent parameters and the finite (usually small) set of propagating energy modes as the dependent parameters. To utilize the ill-posedness in the inverse scheme, we will choose the profile that is more suitable for particular needs. As was mentioned in the introduction, fabrication techniques rely heavily on mechanized processes like laser etching. A fine precision exists to create shape profiles

for diffraction gratings, but ultimately, it has a discernible limit, and after this limit is reached, details in the profile may not be reproducible. Hence, simpler models are more desirable.

To illustrate just how ill-posed the problem is, we will consider a as a piecewise-constant defined function on the finite-element grid. Starting with two differing (significantly different) profiles, we analyze how the profiles will individually converge to a profile that yields the same reflection and transmission modes. The two diagrams below (Figure 3) show an initial profile versus the iterated solution, converging to $J(a) < 10^{-10}$ to a given target distribution of propagating modes. Notice how the final profile maintains the essential features of the initial profile.

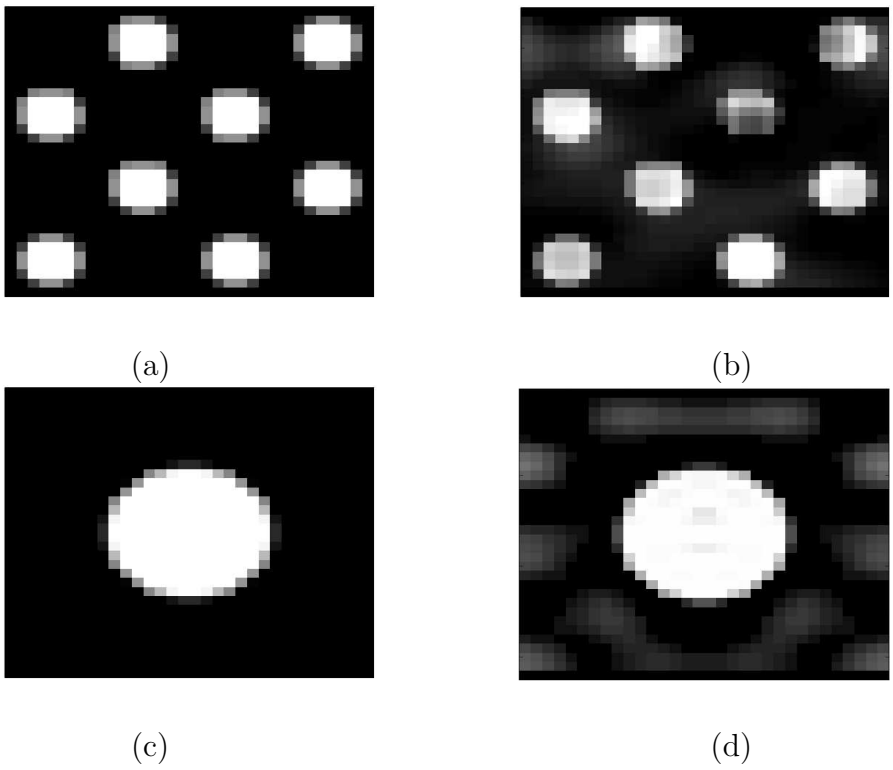


Fig. 3. (a) Initial profile 1; (b) Final, iterated profile 1; (c) Initial profile 2; (d) Final, iterated profile 2.

Accompanying the figure is Table 1 giving the initial energy distribution with corresponding profile. The initial profiles are different. The table shows how the initial reflectance and transmissions are significantly different between profile 1 and profile 2. However, they converge to the same target reflections and transmissions, which is $.3, .2, .2$ for the reflectance, and $.1, .1, .1$ for the transmission.

Table 1. Initial modes for profiles 1 and 2.

	Ref		
profile 1 (a)	0.2647	0.0003	0.0001
profile 2 (c)	0.0774	0.0530	0.0530
	Trans		
profile 1 (a)	0.7348	0.0000	0.0000
profile 2 (c)	0.1844	0.3161	0.3161

This strongly suggests that the profile space is large enough to provide much freedom in the design process. Creating optimal designs is then a matter of preference. The question to answer is how sensitive is the profile for changes in the target propagation modes.

We will partially answer this by studying a linearization of the cost function. Actually, we compute a linearization of the vector of computed propagation modes (the forward map) as this information is more relevant to the current discussion.

We can view how sensitive data is to the forward problem by considering the linearization of the forward map F as follows: We are interested in the behavior of

the derivative of F, the linear operator, $DF(a) : L^2(\Omega) \to \mathbb{R}^m$.[2] But, in the discrete subspace S^h, we will first look at the discrete operator, $DF(a) : S^h(\Omega) \to \mathbb{R}^m$. Further, the one-to-one correspondence of S^h to $\mathbb{R}^{N_1}$ yields the matrix operator,

$$DF(a) : \mathbb{R}^{N_1} \to \mathbb{R}^m, \tag{3.12}$$

where $N_1 = (M+1) \times N$, the size of nodal space. (We make no attempt to distinguish one operator from another in notation. It will be clear from context which operator space we are using.) It is important to note that $N_1 \gg m$, for this makes an under-determined operator, $DF_{m \times N_1}$. From this we compute a Singular Value Decomposition (SVD) and study the resulting eigenspaces. Let

$$DF(a) = U \Sigma V^T,$$

where U is orthogonal $(m \times m)$, Σ is diagonal $(m \times N_1)$, and V is orthogonal $(N_1 \times N_1)$. Let $V \equiv [v_1, v_2, \ldots, v_{N_1}]$. Since $m \ll N_1$, we note that $DF(a)(\delta a)$ is equivalent to $U\Sigma[a_1, a_2, \ldots, a_m]^T$, where $V[a_1, a_2, a_3, \ldots, a_{N_1}]^T = \delta a$. I.e., the first m components from the decomposition of δa in the basis $[v_1, v_2, \ldots, v_{N_1}]$ is what contributes to $DF(a)(\delta a)$, while the remaining components are ignored. To be more specific, we split $V = [V_1 V_0]$, where V_1 is the first m column vectors of V, and V_0 is the remainder. Then V_0 spans the kernel of DF, while V_1 spans the orthogonal complement.

In this setting, we analyze the decomposition $DF(a)(\delta a)$ using a circle profile for a. Define Γ_c to be the boundary of the circle. Refer back to Figure 3(a). In the figure is shown the profile in the grid. It depicts the edge values with an area averaging technique. From the SVD, you will notice that the kernel space is quite large. In terms of sensitivity, there are many directions the profile can take that induce no change

[2]The formal derivation of the DF operator will be deferred until Chapter VII.

in the forward map, thus, suggesting little sensitivity. The linearized r, t vectors are however quite sensitive to the basis vectors associated with nonzero singular values. The following five profiles from Figure 4 are the first five extracted from the SVD. They are precisely the vectors that span the orthogonal complement of the null space for DF.

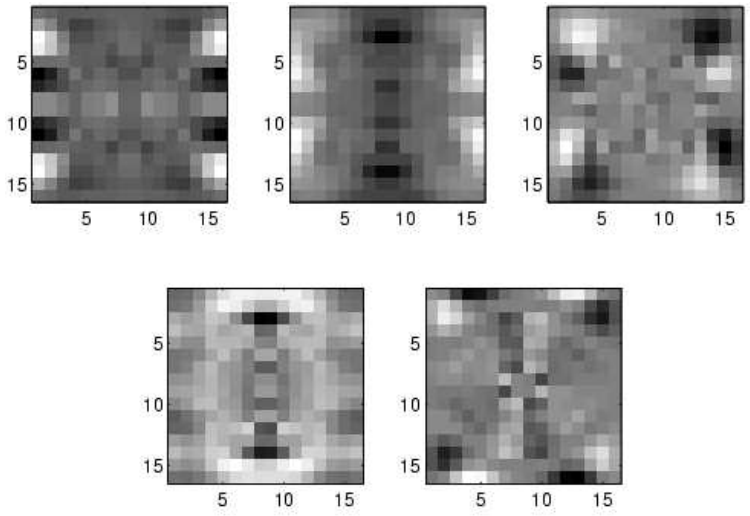


Fig. 4. The 5 profile vectors that affect a change in the DF operator.

Any vector $dk \in \mathbb{R}^{N_1}$ can be decomposed into $dk_1 \oplus dk_0$, where $dk_1 \in V_1$, and $dk_0 \in V_0$. What we analyze here is how much freedom the kernel of DF allows for specific profile changes. The question of interest is whether it is possible to consider profile updates that affect the cost functional in such a way that only the cells that contain the boundary of the circle are used.

F. Constrained Optimization

One way to answer the previous question is to consider a constrained optimization problem. Specifically, we let $I = \{n_1, n_2, \ldots n_t\}$ be the cell indices that do **NOT** intersect Γ_c. Also, define $\tilde{I} = \{1, 2, \ldots, N_0\} - I$ to be the complement indices. From these indices we form the projection subspace $S_I \subset \mathbb{R}^{N_0}$, by restricting to those components, I, and leaving the others (the set $\tilde{I}$) zero. Define the corresponding projection operator $P_I(x)$ as follows:

$$y = P_I(x) \rightarrow \begin{cases} y_k = 0 & \text{for each } k \in \tilde{I}, \\ y_k = x_k & \text{for } k \in I. \end{cases} \tag{3.13}$$

Now, we consider a descent direction for the cost functional. Specifically, consider the gradient, computed as $G = DF^T(F(a) - q)$. Since G may have non-zero components inside the index set, I, we ask if another $\hat{G}$ exists that can affect the cost functional in a descent direction by restricting to the subspace, $S_{\tilde{I}}$. What we will look for is a vector $\hat{G} = G + dk_0$, such that $dk_0 \in V_0$. Define X to be the hyperplane $\{G + V_0 y \mid y \in \mathbb{R}^{N-m}\}$. To solve for $\hat{G}$ requires solving a constrained problem: Find x such that

$$\min_{x \in X} \|P_I(x)\|^2 \tag{3.14}$$

is realized. Clearly, $P_I(x) = 0$ is optimal. If such a solution exists, then it is equivalent to the notion that V_0 is large enough to allow descent directions only along the levelset cells. This in turn will be a measure of the ill-posedness of the problem, for if the boundary cells of an initial profile can alter the cost functional significantly, then little variation from the initial profile will occur to achieve minimization on the cost function. We will explore this more after level-sets are introduced in Chapter V.

Table 2. Comparison of projection of psuedo-inverse solutions for differing k_2.

k_2	$\|P_I(dk_0)\|$	$\|(I - P_I)(dk_0)\|$
2	6.7542e-17	0.2491
2.2	7.1676e-18	0.032356
2.4	1.9871e-17	0.047948
2.6	4.4924e-17	0.17038
2.8	1.1552e-17	0.031579
3	3.0665e-17	0.11928
3.2	1.2943e-16	0.15757
3.4	3.9692e-17	0.13219
3.6	8.0926e-17	0.37421
3.8	2.6289e-17	0.13988
4	9.9631e-17	0.36258

Since this problem is linear, we note the normal equations reduce to $P_I(x) = 0$. We compute a psuedo-inverse on $P_I(G + V_0 y) = 0$, yielding a least squares solution $\hat{y} = -(P_I V_0)^{\dagger}(P_I G)$. Now, if V_0 were full rank then this would give us a unique minimizer, but for this problem, that implies the DF operator is zero. We consider the decomposition of the profile vector $dk_0 = V_0\hat{y} + G$ into $dk_0 = P_I(dk_0) \oplus (I - P_I)(dk_0)$.

For comparison, we ran several profiles (using MATLAB) with varying index of refractions (k_2 varies, but $k_1 = 1$) under the same circular profile of radius 2 (see Table 2). We observe that each run gave a descent direction whos zero projection onto S_I was within machine precision. Thus, the vector dk_0 actually coincides with $S_{\tilde{I}}$. This verifies for us the sensitive nature of computing descent directions within

the initial level-set cells alone, thus causing shapes to not alter significantly from their initial profile.

CHAPTER IV

THE GRADIENT

In this chapter we describe the gradient of the least squares objective (3.10) from the previous chapter. Its properties and development are crucial to the step methods employed in the level-set routines. Formally, a linear operator mapping $L_2 \to \mathbb{C}$, denoted $DJ[a](\cdot)$ is sought. It is extracted from

$$DJ[a](\delta a) = 4\mathrm{Re}\int_\Omega \sum_{n\in\Lambda_0} \frac{\beta_n^0}{\beta^0}\left|\frac{\beta_n^0}{\beta^0}|r_n|^2 - |r_n^*|^2\right| Dr_n[a](\delta a)\overline{r_n[a]} + \sum_{m\in\Lambda_1}\frac{\beta_m^1}{\beta^0}\left|\frac{\beta_m^1}{\beta^0}|t_m|^2 - |t_m^*|^2\right| Dt_m[a](\delta a)\overline{t_m[a]}d\alpha, \tag{4.1}$$

where δa is a small perturbation from a. Thus, the linear operators $Dr_n[a](\delta a)$ and $Dt_m[a](\delta a)$ will be determined using their definitions (3.3). Before we begin, we recall the Helmholtz problem and associated BC's from Chapter II, specifically equations (2.1) to (2.3). The solution can be viewed from the variational setup, and the proof of Theorem 2.6 as the inverse of a map $A : H^1(\Omega) \to H^{-1}(\Omega)$. Here, we define this unique solution as depending explicitly on the profile space, $\mathcal{A} \subset L^\infty(\Omega)$, by defining an operator,

$$\mathcal{F} : \mathcal{A} \subset L^\infty(\Omega) \to H^1(\Omega), \tag{4.2}$$

by $U = \mathcal{F}(a)$. Sometimes we will refer to the operator dependence simply as $U[a]$.

Note that this map is not to be confused with the earlier defined **forward map**, F. They are related, however, as $F(a)$ is the vector of Fourier coefficients of $\mathcal{F}(a)\big|_{\Gamma_0}$ scaled by the propagation mode coefficients, β_n^0 (see (3.9)).

The properties of $\mathcal{F}$ will be investigated in this chapter. First, it was mentioned in Chapter III from Theorem 3.2 that this operator is continuous and uniformly

bounded over all profiles and for a specific range of frequencies. We will prove this as a consequence of the differentiability of the map. We begin by considering the solution δU that satisfies a "linearized" problem,

$$
\begin{aligned}
(\Delta_\alpha + a)\delta U =& -(\delta a)U, \ in\, \Omega, \\
\left(T_j - \tfrac{\partial}{\partial \eta}\right)\delta U =& \ \ 0 \qquad\quad on\, \Gamma_j\,, \ \ j = 0, 1.
\end{aligned} \tag{4.3}
$$

It derives this name from the following observation:

$$
\left\| \widehat{U} - U - \delta U \right\|_{H_1(\Omega)} \leq C \left\| \delta a \right\|_{L_\infty}^2 ,
$$

where $\hat{a} = a + \delta a$, $\widehat{U} = U[\hat{a}]$. Before this is proved let it be known that this establishes Frechét differentiability of the solution operator $\mathcal{F}(a)$ at each point a.

A. Frechét Differentiability

Lemma 4.1 (Frechét Differentiable)

Assume that the profile functions, a and $\widehat{a} = a + \delta a$, satisfy the conditions in Theorem 2.6. Let U be a solution to the variational problem (2.5) with squared index function a, and likewise for $\widehat{U}$ with $\widehat{a}$. Let δU solve the above problem (4.3). Then

$$
\left\| \widehat{U} - U - \delta U \right\|_{H^1(\Omega)} \leq C \left\| \delta a \right\|_{L_\infty}^2 , \tag{4.4}
$$

where the constant C depends only on Ω, a_0, and a_1, the refractive index bounds.

Proof. Define the operator equation $A_a U_a = f$ that represents the variational problem (2.5). In Chapter II the variational form (2.5) was proven to admit a bounded inverse, independent of the shape of the interface. This means there is a $C > 0$ such that for each profile a, $\|A_a^{-1}\| \leq C$. Starting from the equations for each of the solutions, U

and $\widehat{U}$, we subtract to obtain the system

$$
\begin{aligned}
(\Delta_\alpha + a)(\widehat{U} - U) &= -\delta a \widehat{U}, \\
(T_0 - \tfrac{\partial}{\partial \eta})(\widehat{U} - U) &= 0, \\
(T_1 - \tfrac{\partial}{\partial \eta})(\widehat{U} - U) &= 0.
\end{aligned} \tag{4.5}
$$

Let $v \in H^1(\Omega)$. We denote by $B(u, v)$ the variational form:

$$B(u,v) = \int_\Omega \nabla u \cdot \overline{\nabla v} - \int_{\Gamma_0} T_0 u \overline{v} - \int_{\Gamma_1} T_1 u \overline{v} - 2i\alpha \int_\Omega \frac{\partial}{\partial x_1} u \overline{v} - \int_\Omega (a - |\alpha|^2) u \overline{v}.$$

Let $\langle \cdot, \cdot \rangle$ represent the standard L^2 innerproduct over Ω. Then the variational equation is to find $U \in H^1(\Omega)$ such that $B(U, v) = \langle f, v \rangle$ for all $v \in H^1(\Omega)$. If U exists and is unique then we are solving the equation $A_a(U) = f$, where A is an $H^1(\Omega) \to H^{-1}(\Omega)$ invertible operator. The f in this case is $\delta a \widehat{U}$. From Chapter II, the variational form (2.5) is identical to $B(u, v)$, the difference between the problems being the right hand side. Thus, the inverse operator A_a exists and is bounded, independent of the profile a. We then consider

$$\|\widehat{U} - U\|_{H^1} \leq \|A_a^{-1}\| \|f\|_{H^{-1}} \leq C \|f\|_{L^2}. \tag{4.6}$$

(Note: This establishes continuity of $\mathcal{F}$ (4.2), by considering that $\|\widehat{U}\|_{H^1} \leq C_1$, which was proved in Theorem 2.6.) Next, combine (4.5) with the "linearized" problem, (4.3). Let $G \equiv \widehat{U} - U - \delta U$. Then G satisfies the system

$$
\begin{aligned}
\Delta_\alpha(G) + a(G) &= -\delta a(\widehat{U} - U), \\
T_0 G - \frac{\partial G}{\partial \eta} &= 0, \\
T_1 G - \frac{\partial G}{\partial \eta} &= 0.
\end{aligned} \tag{4.7}
$$

That is, G satisfies the same operator equation (4.5) with the same profile a. Thus,

$G = A_a^{-1}(-\delta a(\widehat{U} - U))$, which gives

$$\|G\|_{H^1} \leq \|A_a^{-1}\|\|\delta a\|_\infty \|\widehat{U} - U\|_{H^1} \leq C^2 \|\delta a\|_\infty^2 \|\widehat{U}\|_{H^1} \leq C_2 \|\delta a\|_\infty^2.$$

This completes the proof. □

The expressions $Dr_n[a](\delta a)$ can be calculated from $r_n = \frac{e^{-i\beta_n^0 y_0}}{2\pi} \int_{\Gamma_0} (U \mid_{\Gamma_0}) e^{-inx} dx$ (see (3.3)) in the following way. Due to the linearity that we seek, we consider a linearization of the reflectance through the linear problem (4.3). Namely, let

$$\begin{aligned} Dr_n[a](\delta a) &= \frac{e^{-i\beta_n^0 y_0}}{2\pi} \int_{\Gamma_0} (\delta U \mid_{\Gamma_0}) e^{-inx} dx \text{ and} \\ Dt_m[a](\delta a) &= \frac{e^{+i\beta_m^1 y_1}}{2\pi} \int_{\Gamma_1} (\delta U \mid_{\Gamma_1}) e^{-imx} dx. \end{aligned}$$

Similar to the Frechét inequality above, (4.4), we note that

$$|\widehat{r_m} - r_m - Dr_m|^2 \leq c \int_{\Gamma_0} \mid \widehat{U} - U - \delta U \mid^2 \leq C \left\| \widehat{U} - U - \delta U \right\|_{H^1(\Omega)}^2$$

by virtue of the Trace theorem. Thus, by uniqueness of the derivative (the Frechét condition, [23]), this establishes the derivatives for the reflectance and transmission coefficients.

In designing methods for optimal shapes, we require a gradient useful for step directions in the profile space. So, far, the above defined derivatives are not satisfactory for this purpose. What is required is an element (vector) in the design space that allows for proper step directions. This can be realized by seeking an operator of the form $Dr_n[a](\delta a) = \langle \delta a, v \rangle_{L^2(\Omega)}$. The element v is then in $L^2(\Omega)$.

B. Adjoint Equation

To resolve this, consider an adjoint problem. First, we introduce a new complex number, ψ, which will relate the adjoint spaces. Let

$$w_m^0 \equiv w[m, a, \alpha, \psi, 0] \tag{4.8}$$

solve the problem,

$$\begin{aligned} (\Delta_\alpha + a) w_m^0 &= 0, \ \ in\, \Omega, \\ \left(T_0^* - \tfrac{\partial}{\partial \eta}\right) w_m^0 &= -\psi e^{imx}, \ \ on\, \Gamma_0, \\ \left(T_1^* - \tfrac{\partial}{\partial \eta}\right) w_m^0 &= 0, \ \ on\, \Gamma_1. \end{aligned} \tag{4.9}$$

Note the dependence on the suppressed parameters. This is necessary due to the dependence of the index m with Λ_j, j=0,1.

The problem is called adjoint because the operators T_j^*, j=0,1, are the adjoint operators when considered as the respective L_2 adjoint on Γ_j, j = 0,1. Specifically, we consider $\langle T_j u, v\rangle_{\Gamma_j} = \langle u, T_j^* v\rangle_{\Gamma_j}$, j=0,1 where

$$T_j^* f = \sum_{n \in Z} -i\overline{\beta_n^j} f_n e^{inx}.$$

Similarly, we will also consider the solution $w_m^1 \equiv w[m, a, \alpha, \psi, 1]$ to the problem

$$\begin{aligned} (\Delta_\alpha + a) w_m^1 &= 0, \ \ in\, \Omega, \\ \left(T_0^* - \tfrac{\partial}{\partial \eta}\right) w_m^1 &= 0, \ \ on\, \Gamma_0, \\ \left(T_1^* - \tfrac{\partial}{\partial \eta}\right) w_m^1 &= -\psi e^{imx}, \ \ on\, \Gamma_1. \end{aligned} \tag{4.10}$$

We use the notation w without subscripts when we are not specifying which boundary, nor which mode the function may represent. Then we can view w as a generic linear

combination of such functions. This is primarily for ease of notation. Now, we consider the variational forms of problems (4.3) and (4.9) (or, symmetrically, (4.3) with (4.10)). By substituting in particular functions $v = w$ and $v = \delta U$ the equations read

$$\begin{aligned}(4.11)\quad \int_\Omega \nabla\delta U\cdot\overline{\nabla w} - 2i\alpha\int_\Omega \partial_1\delta U\overline{w} - \int_\Omega aw\overline{w} + \alpha^2\int_\Omega \delta U\overline{w} \\ - \int_{\Gamma_0} T_0\delta U\overline{w} - \int_{\Gamma_1} T_1\delta U\overline{w} &= \int_\Omega (\delta a)U\overline{w},\end{aligned}$$

$$\begin{aligned}(4.12)\quad \int_\Omega \nabla w\cdot\overline{\nabla\delta U} - 2i\alpha\int_\Omega \partial_1 w\overline{\delta U} - \int_\Omega aw\overline{\delta U} + \alpha^2\int_\Omega w\overline{\delta U} \\ - \int_{\Gamma_0} T_0^* w\overline{\delta U} - \int_{\Gamma_1} T_1^* w\overline{\delta U} &= \int_{\Gamma_0}\psi e^{imx}\overline{\delta U}.\end{aligned}$$

Using integration by parts we have that $\int_\Omega \partial_1\delta U\overline{w} = -\int_\Omega \partial_1\overline{w}\delta U$. And since $\langle T_j u, v\rangle_{\Gamma_j} = \langle u, T_j^* v\rangle_{\Gamma_j}$ we notice that by taking the conjugate of the expressions in (4.11) and subtracting (4.12) yields

$$(4.13)\qquad \overline{\psi}\int_{\Gamma_0}\delta U e^{-imx} = \int_\Omega (\delta a)U\overline{w}.$$

Recall, we are looking for a linear operator $Dr_n[a] : L_2 \to \mathbb{C}$. We have, by the right hand side of (4.13), an L_2 gradient of $Dr_n[a](\delta a)$, provided we solve $w_n^0 \equiv w[n, a, \alpha, \psi, 0]$. Thus,

$$(4.14)\qquad Dr_n[a](\delta a)\overline{\psi} = \frac{e^{-i\beta_n^0 y_0}}{2\pi}\overline{\psi}\int_{\Gamma_0}(\delta U\mid_{\Gamma_0})e^{-inx}dx = \frac{e^{-i\beta_n^0 y_0}}{2\pi}\int_\Omega (\delta a)U\overline{w_n^0},$$

which is the adjoint representation, $\langle Dr_n[a](\delta a), \psi\rangle = \langle \delta a, Dr_n[a]^*(\psi)\rangle_{L_2}$. Similarly, $Dt_m[a](\delta a)$ has an associated L^2 inner product formulation with corresponding function $w[m, a, \alpha, \psi, 1]$.

Finally, we can construct the derivative,

$$DJ[a](\delta a) = 2Re\int_\Omega \delta a U_\alpha\left\{\sum_n \overline{\frac{e^{i\beta_n y_0}}{2\pi}w_n^0} + \sum_m \overline{\frac{e^{-i\beta_m y_1}}{2\pi}w_m^1}\right\},$$

where the notation $\overline{w_n^j}$ is defined by (4.8). We rewrite this with explicit operator $L_2(\Omega) \to \mathbb{C}$,

$$DJ[a](\delta a) = 2Re \int_\Omega \delta a U_\alpha \overline{\left\{ \sum_{n\in\Lambda_0} \frac{e^{i\beta_n y_0}}{2\pi} w_n^0 + \sum_{m\in\Lambda_1} \frac{e^{-i\beta_m y_1}}{2\pi} w_m^1 \right\}} d\Omega.$$

We can simplify this functional form by considering the problems (4.9) and (4.10) are linear. Thus, a linear combination of solutions w_n^0 and w_m^1 will satisfy the same PDE, but with an appropriate linear combination of BC's. Let

$$W \equiv \sum_{n\in\Lambda_0} \frac{e^{i\beta_n y_0}}{2\pi} w_n^0(x,y) + \sum_{m\in\Lambda_1} \frac{e^{-i\beta_m y_1}}{2\pi} w_m^1(x,y).$$

W solves

$$\begin{aligned} (\Delta_\alpha + a)W &= 0, \ in\, \Omega, \\ \left(T_0^* - \tfrac{\partial}{\partial\eta}\right) W &= -\sum_n r_n[a] \frac{e^{i\beta_n^0 y_0}}{2\pi} e^{inx}, \ on\, \Gamma_0, \\ \left(T_1^* - \tfrac{\partial}{\partial\eta}\right) W &= -\sum_m t_m[a] \frac{e^{-i\beta_m^1 y_1}}{2\pi} e^{imx}, \ on\, \Gamma_1, \end{aligned} \tag{4.15}$$

then

$$DJ[a](\delta a) = 2Re \int_\Omega \delta a U_\alpha \overline{W} d\Omega. \tag{4.16}$$

We thus identify the function $U_\alpha \overline{W}$ with the gradient of J. The gradient in the multi-layered case will be deferred until the chapter on multi-layers. We end this chapter with a statement of the regularity of the gradient.

Theorem 4.2

The gradient, $U_\alpha \overline{W}$, as defined above, lies in $H^1(\Omega)$.

Proof. Recall from Chapter II, Theorem 2.6. There, it was established that U_α and W lie in $H^2(\Omega)$. From Schauder's Lemma [5] it follows that $U_\alpha \overline{W}$ lies in $H^2(\Omega)$, as

well. Therefore, $U_\alpha \overline{W} \in H^1(\Omega)$. $\square$

CHAPTER V

MULTI-LAYERED STRUCTURES

In the previous chapter, we analyzed the properties of the gradient observed as a single layer system. This chapter will discuss the multi-layered structure and how it relates to a single layer setup.

A. Repeated Parameters

We consider a structure composed of several identical layers. First, we describe the independent parameters. We define the profile space $\mathcal{A}$ as $\{k(x_1, x_2) \in L_\infty(\Omega_1)| \quad k_1 \leq k \leq k_2, \quad 0 \geq x_2 \geq -d\}$. The repetition of the layered structure is equivalent to the repetition of the parameters as we traverse the layers. That is,

$$k(x_1, y) = k(x_1, x_2), \quad y + jd = x_2,$$

where $-d \leq y + jd \leq 0$ (see Figure 5).

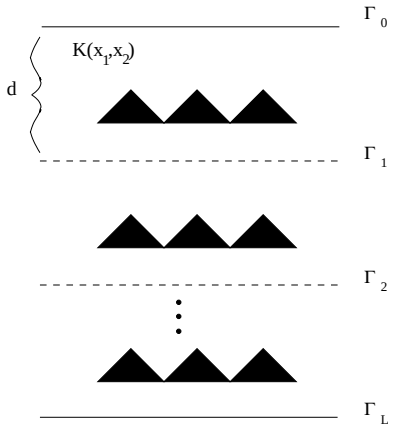


Fig. 5. Multilayered setup.

To see the interplay with the dependent and independent parameters, we consider a simple example of a function from R^3 to R^1, $f(x_1, x_2, x_3)$. The directional derivative of the function, $Df(\hat{h})$, where $\hat{h} = \langle h_1, h_2, h_3 \rangle^T$ is the direction is given as

$$\langle \frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \frac{\partial f}{\partial x_3} \rangle \langle h_1, h_2, h_3 \rangle^T.$$

Now, define the function $g(t) : R^1 \rightarrow R^1$ by $g(t) = f(t, t, t)$. The derivative reduces to $\sum_i \frac{\partial f}{\partial x_i}$. In a like manner, in our structure all layers are identical, so the parameter space $k(x_1, y)$, where $-Ld \leq x_2 \leq 0$, is reduced to $k(x_1, x_2)$.

B. Multi-Layered Gradient

In Chapter IV, we presented a general procedure for the gradient. The gradient in the multi-layered case follows through with no hitches. Recall the gradient in its final L_2 form is (see (4.16))

$$DJ[a](\delta a) = 2Re \int_\Omega \delta a U \overline{W} d\Omega.$$

With our specific model of a repeat in the layered structure, we have to keep our profile differential, δa, in the same space, $\mathcal{A}$. This means that it is the function

$$\delta a(x_1, x_2) = \{h(x_1, x_2 + j \cdot d),\ -d < x_2 + jd \leq 0,\ 0 \leq j < N,$$

where $h \in L^\infty(\Omega_0)$. Therefore, under a simple change of variable in the second component, we can finalize the gradient as

$$\begin{aligned} \langle \delta a, G \rangle = 2Re \int_\Omega \delta a G &= 2Re \sum_j \int_{\Omega_j} h(x_1, x_2 + jd) G \\ &= 2Re \int_{\Omega_1} h(x_1, y_2) \sum_j G(x_1, y_2 - jd) dx_1 dy_2. \end{aligned} \tag{5.1}$$

This is an average of the original (one layer) gradient along the vertical direction:

$$G = \sum_j U(x, y - jd)\overline{W(x, y - jd)}$$

Hence, the only difference between the single layer gradient and the multilayer gradient is the summation over the repeated layers.

CHAPTER VI

LEVEL SETS

The Level-Set Method has been used successfully in a number of applications [39] ranging from circuit design to optimal shape constructions in stress models to propagating wave fronts. Its power lies in its ability to track motion in a numerically stable fashion being based on equations developed from conservation laws in gas dynamics. The Level-Set Method was developed as an alternative to front tracking, which depends on systematically tracing the paths of specialized markers and letting it evolve under a set of equations. The common problem with front tracking is that motion around corners and cusps allow for error to build substantially, and, therefore, is unstable. Another feature (or lack thereof) is that marker methods do not allow for new regions to form. This is an inherent feature under level-sets. In an optimal design setting, level-sets have been used for creating optimal structural designs [40, 27]. In this paper, we consider optimal designs by using the Level-Set Method as outlined in [34].

From the introduction, we mentioned a level-set, being the zero-contour curve from a continuous surface profile, $z = \phi(x)$. Formally, the level-set is defined by

$$\text{level-set} \equiv \{x \in \Omega : \phi(x) = 0\}.$$

Using the Implicit Function Theorem [33], we know that this set is continuous, and depending on the regularity of the profile, has sufficient smoothness. The level-set divides Ω into two disjoint regions, Ω_+, where $\phi(x) \geq 0$, and Ω_-, where $\phi(x) < 0$. Since $\phi(x)$ is periodic in x_1, the level-set regions can extend (wrap) in that direction. We define the dielectric medium as follows: In the individual region(s) comprising Ω_+ we associate with the medium of refractive index k_2, and likewise, we assign k_1 to the

region(s) comprising Ω_-. In the implementation of the level-set [1], we will be more concerned with the squared refractive index, $a(x) \equiv \omega^2 k(x)^2$. So, we define $a_1 = \omega^2 k_1^2$ and $a_2 = \omega^2 k_2^2$. We define the Squared Index Profile function as follows:

Definition 6.1 (Squared Index Profile function with level-set)

$$a(x) = \begin{cases} a_2, & \phi(x) \geq 0, \\ a_1, & \phi(x) < 0 \end{cases}. \tag{6.1}$$

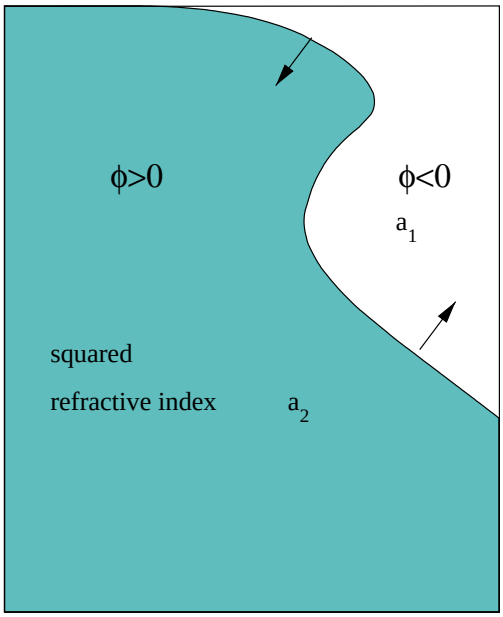


Fig. 6. A representation of a level-set.

[1]A word of usage : The level-set has two distinctions. It is properly a set in Ω. In certain situations we refer to the Squared Index Profile function as synonymous with the level-set. The level-set changes by a spatial variation δx on Ω. The Squared Index Profile function cannot be described as succinctly, being in $L^\infty(\Omega)$. However, there is no confusion when described in context.

To allow the level-set to change by δx, it is necessary to trace the corresponding change in surface profile, $\delta\phi$. Formally, (see [34]) we consider that the change in the surface function must satisfy a direct corresponding change in the level-set. We represent a linearization of this change. We assume that $\hat{a} = a+\delta a$ represents the new squared index profile function derived from the updated surface function $\hat{\Phi} = \phi + \delta\phi$. As depicted in Figure 6, we further assume that δx moves perpendicular to the level-set. Let $\boldsymbol{\eta}(x) = \dfrac{\nabla\phi(x)}{|\nabla\phi(x)|}$ be the normal. A governing equation relates the surface profile ϕ to the change in x:

$$0 = \hat{\Phi}(x+\delta x) = \hat{\Phi}(x) + \nabla\hat{\Phi}\cdot\delta x. \tag{6.2}$$

But, $\hat{\Phi} = \phi + \delta\phi$. Substitution into the above equation and using $\phi(x) = 0$ due to being a level-set yields

$$\delta\phi + \nabla\phi\cdot\delta x + \nabla\delta\phi\cdot\delta x = 0.$$

Now, ignoring the higher order term yields the following equation:

$$\delta\phi = -\nabla\phi\cdot\delta x. \tag{6.3}$$

Next, a changes by a fixed quantity in the local area around x. Referring to Figure 7, we consider moving the level-set locally from x to $x+\delta x$. In regions where the level-set moves in the direction of the normal, $\boldsymbol{\eta}(x)$, the region is now outside the level-set, and the refractive index is a_1. Thus, δa changes by $-(a_2 - a_1)$. For directions $-\boldsymbol{\eta}(x)$, δa changes by $(a_2 - a_1)$. Next we quantify, in some sense, the variation of the change in profile index function. Infinitesimally, the change is proportional to $(a_2 - a_1)$, dependent on the size of δx. To make precise our sets, we refer to Figure 7 again. We define D to be the region associated with a_2 and has boundary $C = \partial D$. Then under a small variation of the level-set due to $\delta\phi$, we denote the new region D' associated

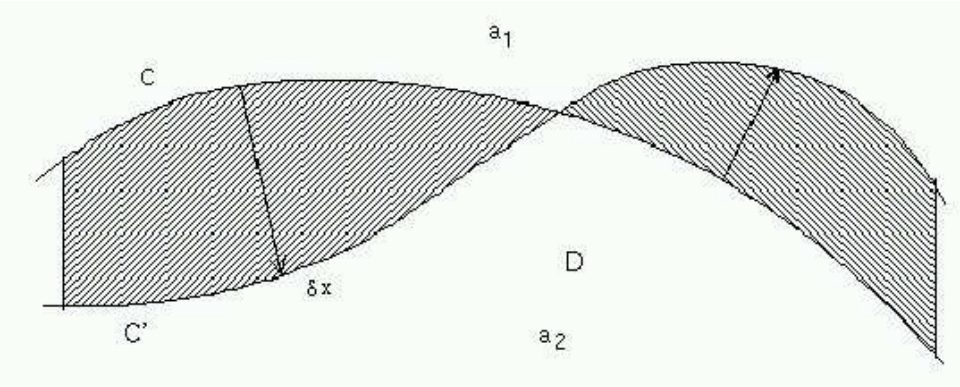


Fig. 7. Moving level-set interface.

again with a_2 but with boundary C'. We consider the changed areas, denoted by a symmetric difference,

$$\tilde{D} \equiv (D - D') \bigcup (D' - D),$$

which includes both the regions where the level-set expands and recedes. Figure 7 depicts this region in the shaded parts.

We define a measure. First, we define the function $\zeta(x)$ by

$$\delta x = \zeta(x)\boldsymbol{\eta}(x). \tag{6.4}$$

We postpone a discussion of the regularity of $\zeta(x)$ until later. By considering an inner product on the set $\tilde{D}$, we look at $\langle \delta a, f \rangle_{\tilde{D}}$ defined by $\int_{\tilde{D}} \delta a(x) f(x)$, where the integral is defined in the sense of Lebesgue, and f and $\delta a \in L^2(\Omega)$.

Before we analyze this function further, we consider a more general framework for integrals over a 2-D region. Consider Figure 8. The region on the left depicts a typical region between the two level-set boundaries, like C and C' above. The region on the right represents a transform under a smooth, one-to-one map, Σ, such that one

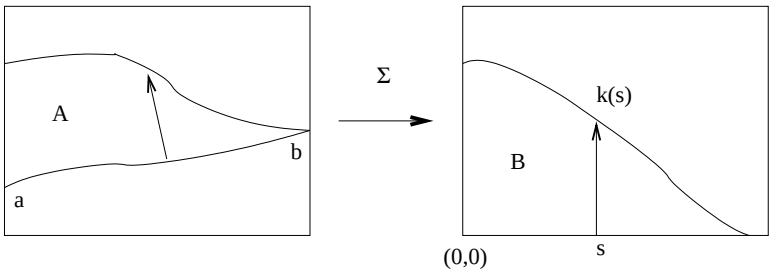


Fig. 8. Transformation of bounded level-set region.

boundary is "straightened". Under a change of coordinates, $\int_A f(x)dx = \int_B \hat{f}(z)dz$, where $z = \Sigma^{-1}(x)$. If f is continuous and the path C is smooth, we consider the integral over the region B as depicted in Figure 8. Specifically, under an arc-length parameterization $s(x)$ of C between a and b, we let $s(a) = 0$ and $s(b) = \text{length}(C)$. We consider the integral $\int_0^{s(b)} \int_0^{n(s)} \hat{f}(s,y)ds\,dy$. Applying the Mean Value Theorem for Integrals, we define the function $0 \leq t(s) \leq n(s)$ such that

$$\hat{f}(s,t(s))n(s) = \int_0^{n(s)} \hat{f}(s,y)dy.$$

The integral then reduces to $\int_0^{s(b)} \hat{f}(s,t(s))n(s)ds$.

Now, by applying the above general technique, we equate the integral $\int_{\tilde{D}} \delta a(x) f(x)$ with

$$\int_C (a_2 - a_1) f(h(x))\zeta(x)ds(x), \tag{6.5}$$

where $\zeta(x)$ is given by (6.4). $s(x)$ is the arc-length along C, and $h(x)$ takes on the same role as $t(s)$ as defined above. A measure of the change in the level-set is thus defined, namely,

$$m(\delta\phi) = (a_2 - a_1)\zeta(x)ds(x).$$

The sign of $\zeta(x)$ is dependent on how the level-set is realized. Since $\frac{\nabla\phi}{|\nabla\phi|}$ as oriented into the refractive region of higher index, a_2, $\zeta(x) > 0$ yields δx directed into D as depicted in Figure 7. This corresponds to δa decreasing by $a_2 - a_1$. I.e., the level-set shifts into the old region occupied with the higher index of refraction and, thus, reduces it to the lower index of refraction . This defines how δa relates to $\zeta(x)$:

$$\delta a(x) = -(a_2 - a_1)\zeta(x). \tag{6.6}$$

Thus, the rule is $\delta\phi$ increasing corresponds with δa increasing, whereas the level-set moves in direction $-\boldsymbol{\eta}(x)$.

A. Decreasing the Cost Functional

We consider how to decrease the cost functional. The main concern is to what direction, δa, does the cost decrease. Clearly the directional derivative $\langle DF, \delta a\rangle_{L^2}$ determines this. The goal is to describe a δa that computes a descent direction. From equations (6.3) and (6.6), the relation between a differential change in ϕ and a is through the function $\zeta(x)$. Specifically, since

$$\delta\phi = \quad -\nabla\phi \cdot \delta x,$$

and

$$\delta x = \zeta(x)\frac{\nabla\phi(x)}{|\nabla\phi(x)|},$$

we have

$$\delta\phi = \ -|\nabla\phi|\zeta(x). \tag{6.7}$$

So far, the functions have been restricted to the boundary $C \equiv \partial D$. Given a descent direction $h(x) \in H^1(\Omega)$, the restriction $h|_C$ (for C Lipschitz) generates an $H^{\frac{1}{2}}(C)$ function for a profile update,

$$\zeta \equiv \frac{h|_C}{-(a_2 - a_1)}.$$

With this definition for ζ, we can construct any number of schemes to produce optimal level-sets. Notice how this restricted function allows freedom for what happens away from the level-set. Indeed, due to continuity, the variation of ϕ sufficiently far away from the level-set does not interfere with what is happening locally at the level-set. To compute a proper descent update for the cost function, we consider global functions on Ω. The natural choice for extensions of ζ is to keep the function h. Ultimately, however, this will affect how the level-set evolves, for as is depicted in (6.7), the gradient of ϕ is used for an update. Thus, how ϕ changes away from the level-set determines much of the evolution. Indeed, it is this feature that makes this optimal design strategy so intriguing.

B. Descent Step

A couple of different descent strategies will be studied for the optimization method. First, we consider a simple gradient descent direction. This provides for

$$\zeta = \frac{-G|_C}{-(a_2 - a_1)},$$

where G is the gradient defined in (4.16). The step update for ϕ is

$$\phi_{n+1} = \phi_n - |\nabla \phi_n| \frac{G}{(a_2 - a_1)}. \tag{6.8}$$

Second, we will also analyze a Gauss-Newton method for comparison [9]. Under

this approach, we consider the descent direction, $\mathbf{Z}$ computed as

$$DF^*DF\mathbf{Z} = -DF^*(F - \mathbf{q}),$$

where DF is the differential operator on F, defined in Chapter III, (3.9), which is a hybrid Newton-gradient step, also known as the Gauss-Newton step. Then this provides for

$$\zeta = -\frac{\mathbf{Z}|_{\mathbf{C}}}{(a_2 - a_1)}.$$

In the chapter on computation, Chapter VII, we will discuss its implementation and investigate convergence through numerical experiments.

CHAPTER VII

COMPUTATION

In this chapter we consider the computational aspects of employing the level-set methods. In Chapter II, we considered the finite element scheme employed to solve the Helmholtz problem (2.5). Here we provide an overview of the various techniques to solve for an optimal refractive shape (the inverse problem) in the paradigm of level-sets. The surface profile function, $z = \phi(x)$, from which the level-set is formed, is implemented as a piecewise planar function on each rectangular cell by splitting the cells into two similar triangles across the diagonals. Except for the nodal points, this approach allows for an easily computable gradient, which is necessary for the evolution step.

A. Evolution Step

Recall from Chapter VI the evolution step employed to update the surface profile function. Explicit in its formulation was the use of the gradient. In Chapter IV we discussed the gradient and how it is expressed as a linear operator on L_2. Recall the cost functional, $J(a)$:

$$J(a) = \|F(a) - \mathbf{q}\|^2,$$

as first described in Chapter III, (3.10).

One step method used is the Gauss-Newton step [9]. We study the operator F, since we are concerned with computing the linear operator DF^*DF. That is, we solve the system, $DF^*DFx = -DF^*(F - \mathbf{q})$. First, it should be emphasized that this determines the normal equations. To compute it efficiently, we break it apart with two separate PDE solves, as will be described in what follows.

In the continuous operator description, DF is given component-wise as

$$DF(a)(\delta a) = 2\mathrm{Re}\Big\{ \big[\frac{\beta_n^0}{\beta^0}\left(\frac{\beta_n^0}{\beta^0}|r_n|^2 - |r_n^*|^2\right)\overline{r_n} Dr_n[a]\delta a\big], \tag{7.1}$$

$$\big[\frac{\beta_n^1}{\beta^0}\left(\frac{\beta_n^1}{\beta^0}|t_n|^2 - |t_n^*|^2\right)\overline{t_n} Dt_n[a]\delta a\big]\Big\}. \tag{7.2}$$

To compute $DF(\delta a)$, we appeal to the adjoint equations defined in Chapter IV, (4.14). There, the identification is made:

$$Dr_n[a](\delta a)\overline{C_n} = 2\pi E_n \overline{C_n}(\delta U)_n = \int_\Omega \delta a U \overline{W_n}, \tag{7.3}$$

where $E_n = \frac{e^{\pm\beta_n^j y_j}}{2\pi}$, $j = \{0,1\}$, δU solves the linearized problem, (4.3), and W_n solves the PDE,

$$\begin{aligned}
(\Delta_\alpha + a)\, W_n &= 0, && \text{in } \Omega,\\
(T_0^* - \frac{\partial}{\partial \nu})W_n &= -\psi_n e^{inx}, && \text{on } \Gamma_0,\\
(T_1^* - \frac{\partial}{\partial \nu})W_n &= 0, && \text{on } \Gamma_1,
\end{aligned}$$

and $\psi_n = \overline{E_n} C_n$. In this case, we set

$$C_n = 2\frac{\beta_n^0}{\beta^0}\left(\frac{\beta_n^0}{\beta^0}|r_n|^2 - |r_n^*|^2\right) r_n.$$

We note here that the subscript notation used above refers to the indices corresponding to the propagating modes, from the sets, Λ_0 and Λ_1. To simplify which element we refer to, we let the components of the vector $\mathbf{v} \in \mathbb{R}^m$ refer to the corresponding mode as shown in the ordered enumeration, $[1 \dots m] \to \{\Lambda_0 \bigcup \Lambda_1\}$. In addition, since the system is linear (refer to the description in Chapter IV on equation (4.15)) we define a linear operator A to relate the solution W_n described above from the coefficients ψ_n

to read as $W = A(\mathbf{v}) : \mathbb{R}^m \to H^1(\Omega)$. Here W solves

$$\begin{aligned}
(\Delta_\alpha + a)\, W =& 0, && \text{in } \Omega,\\
(T_0^* - \frac{\partial}{\partial \nu})W =& -\sum_{m\in\Lambda_0} v_m e^{imx}, && \text{on } \Gamma_0,\\
(T_1^* - \frac{\partial}{\partial \nu})W =& -\sum_{m\in\Lambda_1} v_m e^{imx}, && \text{on } \Gamma_1.
\end{aligned}$$

Now, we carefully pick out the appropriate DF and DF^* operators from the above expressions. First, we note the linear operator DF maps $DF : L^2 \to \mathbb{R}^m$. From the last equality in (7.3), we identify $\langle \mathbf{v}, DF(\delta a)\rangle_{\mathbb{R}^m}$ as $\sum E_m \overline{C}_m (\delta U)_m \overline{v}_m$. By the definition of an adjoint, we define the operator DF^* to be $U\overline{W}$, such that $\langle DF^*\mathbf{v}, \delta a\rangle_{L^2(\Omega)}$ will correspond to the right hand expression in (7.3), and where the vector $\mathbf{v} \in \mathbb{R}^m$ is the collection of coefficients used for the modes in the propagating waves, $W = A(\mathbf{v})$. Thus, letting $v_m \equiv E_m \overline{C_m}(\delta U)_m = DF(\delta a)_m$, we have that

$$\langle DF^*(DF(\delta a)), \delta a\rangle = \sum |v_m|^2 = \langle DF(\delta a), DF(\delta a)\rangle_{\mathbb{R}^m},$$

proving that the operator DF^*DF is Hermitian and defines the adjoint properly.

In the process of computing DF^*DF, we refer back to equation (7.3) to see that three solves are required. The first is to compute δU in the linearized problem (4.3), the second is the adjoint problem $W = A(\mathbf{v})$ as described above, and the third is, of course, U in the forward problem.

A careful point is made here to distinguish the continuous operator from the discrete. Specifically, the continuous operators $Dr_n[a](\delta a)\overline{C_n}$ and $Dt_n[a](\delta a)\overline{C_n}$ correspond with the continuous functions $U\overline{W_n^0}$ and $U\overline{W_n^1}$, respectively, as depicted in above paragraph. However, we note that the discrete analog is a vector in $\mathrm{R}^{\hat{N}}$, for some $\hat{N}$. Recall, the N_1 denotes the nodal points (see Chapter III). The function $U\overline{W_n}$ is realized as a product of piecewise bilinear functions. The problem is how

to identify the discrete DF operator. There are two separate discrete realizations of $DF = \left[\frac{\partial F_i}{\partial a_j}\right]_{mx\hat{N}}$ to consider depending on $\hat{N}$. In Chapter III, we viewed $\hat{N} = N_1$, from the nodal space of the finite subspace S^h (3.12). Numerically, this linear operator is of the form $\delta a \in \mathrm{R}_1^N \rightarrow \mathrm{R}^m$. Another realization of DF can be viewed as $\mathrm{R}^{N_0} \rightarrow \mathrm{R}^m$, where N_0 is the number of rectangular cells $(M \times N)$. This notion of DF represents the mapping of piecewise constant a in the squared profile space. We view this operator description for the following reason: Consider the identification again from Chapter IV, (4.14). If we define the profile as piecewise constant over a single rectangular element, Ω_k, then,

$$\int_{\Omega_k} \delta a_k U\overline{W_n} = \delta a_k \int_{\Omega_k} U\overline{W_n}, \quad 1 \leq k \leq N_0, \tag{7.4}$$

where N_0 is the number of elements (cells). Thus, taking the average of $U\overline{W_n}$ over the element is the natural value to associate for the discrete analog. This essentially is an operator that takes the product of two functions $f \in S^h(\Omega)$ to a piecewise constant function in $\mathbb{R}^{N_0}$. Since the functions U and W_n are elements in the finite element subspace, the average is computed directly from the nodal points, specifically, the average is given by $U|_{\Omega_k} \rightarrow \mathbf{u} = [u1, u2, u3, u4]$ and $W|_{\Omega_k} \rightarrow \mathbf{w} = [w1, w2, w3, w4]$, and the local mass matrix M as defined by

$$M_{i,j} = \int_{R_n} \Phi_i(x, y)\Phi_j(x, y) dR_n,$$

where $1 \leq i, j \leq 4$, R_n is rectangle in Ω. Thus, the average is computed as $\mathbf{u}M\mathbf{w}^T$.

B. Interpretation of the Level-Set

In light of the above discussion, the Squared Index Profile function a in the computational scheme has different interpretations. The simplest structure is to define it as

piecewise constant. This coincides nicely with the gradient-adjoint formulation given above (7.4). Applying this to the level-set method requires that the value in a cell where the level-set intersects have a value between a_1 and a_2. This is necessary to maintain continuity, for better use in the gradient. The simplest strategy is to assign it a weighted average of the areas of the intersection. But numerical experiments have shown this to fail to produce a descent direction after even a few iterations. In the variational form (2.5) the index is used in the mass matrix $\int_\Omega au\overline{v}$. The question is whether there exist a value a_n on cell R_n, such that

$$\sum_n a_n \int_{R_n} U\overline{V} = \int_\Omega aU\overline{V}.$$

The answer is negative.

Consider Figure 9. The level-set breaks a cell, R, into two distinct regions, R_1, and R_2. The value of a is ill-defined in the following sense:

$$a \int_R U\overline{V} = a_1 \int_{R_1} U\overline{V} + a_2 \int_{R_2} U\overline{V}. \tag{7.5}$$

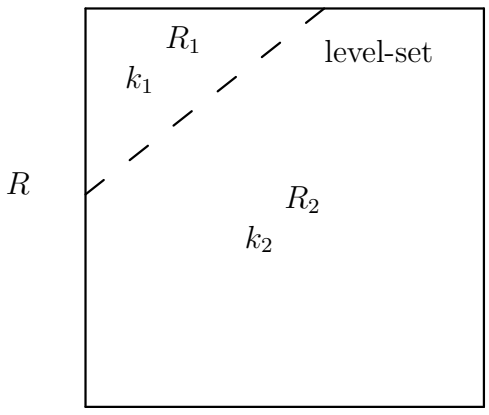


Fig. 9. Typical cell with intersecting level-set.

What is required for this to be true is that each basis pair $\Phi_i\Phi_j$ must satisfy the above equality (7.5) for the cell R. Unfortunately, this is not the case. Due to the different symmetries, the integrals will all have differing values.

Thus, one idea is to find a minimal a that is closest to all 16 basis pairs, in the least squares sense. This is the average of all 16. However, such a scheme is still an average. And it, too, has shown poor convergence in the optimization schemes. We observe that the averaging process for a converges to the right hand side in (7.5) as $h \to 0$. This follows trivially from the fact that the mass integrals approach zero, $\int_\Gamma \Phi_i\Phi_j \to 0$, as $h \to 0$.

So, to circumvent the issue of an appropriate piecewise constant a for the level-set, directly computing the mass matrix with the level-set is implemented. That is, the mass matrices represent the right hand side of (7.5) exactly.

There is no longer an explicit $a \in R^{N_0}$. This will pose a difficulty for computing a Gauss-Newton step, and will be discussed further on that topic.

C. Convergence Comparison of Average Cell-Wise Constants a, Versus Implicit Exact a

In the implementation of this problem, the two different interpretations of the level-set have been studied. Each has its advantages and disadvantages. First, the piecewise constant implementation is attractive for it lends itself to a cleaner description of the gradient. Its drawback, however, is that it does not work well in the level-set optimization routines. Thus, for analysis purposes, we study the piecewise constant implementation, and for computational routines, we mainly use the implicit scheme.

The reasoning as to why the averaging step breaks down is that the value of the profile in a grid cell is too ambiguous for the level-set update scheme. The value is a

weighted area average of where the level-set intersects the cell, $a = \lambda a_1 + (1 - \lambda)a_2$. For all cells, regardless of size, it is possible to slightly alter the level-set with no change in the average. Another way to state this is that the level-set update lies within the kernel of the averaging scheme. The obvious way to try to ameliorate this phenomenon is to decrease the mesh size. This does not alter the averaging process on the boundaries, but more cells will fall inside the level-set region, thereby giving more discernible shapes and more weight to the constant regions. However, there is a trade-off. Smaller weight may be assigned to the regions, but more of them may tend to balance out the gain.

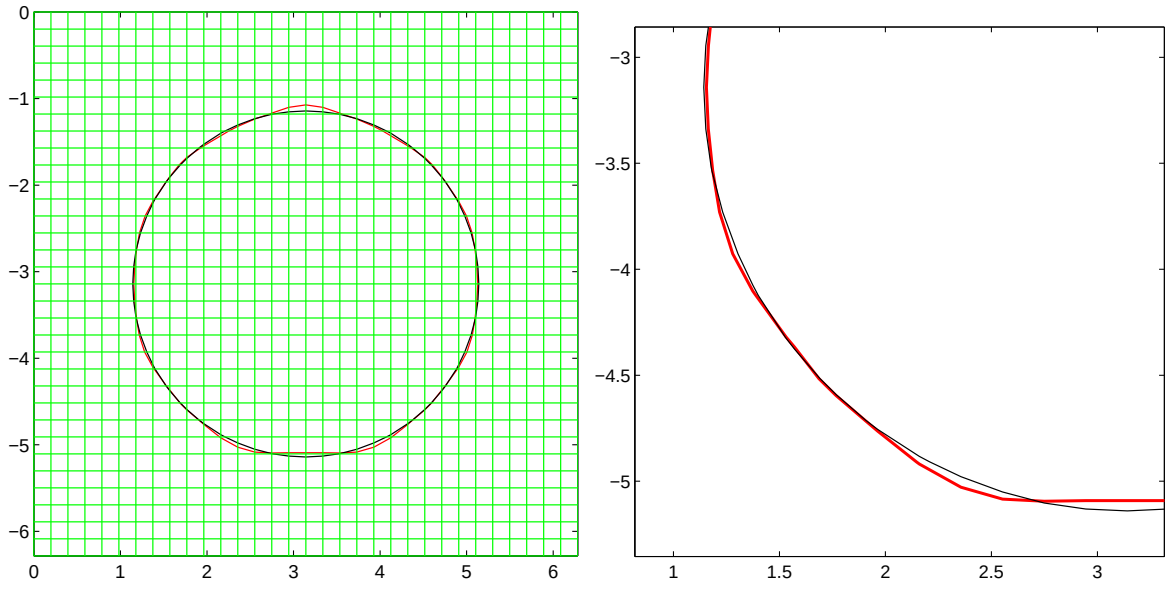
A further approximation occurs with the surface profile update in the optimization scheme. We recall (6.8) and restate it here for convenience,

$$\phi_{n+1} = \phi_n - |\nabla \phi_n| \frac{G}{(a_2 - a_1)}. \tag{7.6}$$

Due to the piecewise constant implementation of the gradient as defined by (7.4), lying in $\mathbb{R}^{N_0}$, and the surface profile function lying in the nodal space $\mathbb{R}^N$, there is some room as to how to make the two compatible with each other. First, we can consider an averaging scheme where we average the gradient around each nodal point's supporting cells. This performs a simple convolution of the gradient around each rectangle to map it into $\mathbb{R}^N$. Another type of approximation is to perform a simple 1-1 map, for example, assigning to each nodal value the lower left quadrant. Then do simple interpolation on the last $N_1 - N_0$ nodes, which will normally be the top or bottom boundary nodes. However, this procedure is not as elegant as the former, and is not pursued further. The point here being that there are many ways to affect the interpolation mapping, and each is a type of approximation, which tends to further stagnate the iteration scheme.

In Chapter III, we discussed the ill-posedness of the problem and how the piece-

wise constant profile can decrease the cost functional narrowly along the boundary cells. Referring to Figure 10, we notice two plots superimposed on each other. The black indicates the initial circular profile, whereas the red indicates the iterated solution, to within 10^{-5}. In the piecewise constant profile, the iterated level-set can not resolve the boundary to achieve adequate convergence. The iteration scheme stagnates.



(a) (b)

Fig. 10. (a) Profile shown with grid. (b) A blowup of the curves on lower quadrant. Sometimes the level-set evolves only within the initial intersecting cells, thereby creating profiles similar in shape to the initial. Typically, in the piecewise constant scheme, this example will fail, due to ambiguity in the descent directions.

D. Barrier at the Interfaces

To maintain the correctness of the problem formulation, the interfaces need to remain "transparent". The index of refraction needs to remain constant in a neighborhood of the boundaries, Γ_0, Γ_1. The level-set as shown in Figure 11 indicates that it may sometimes creep across the interfaces. This violates the problem formulation.

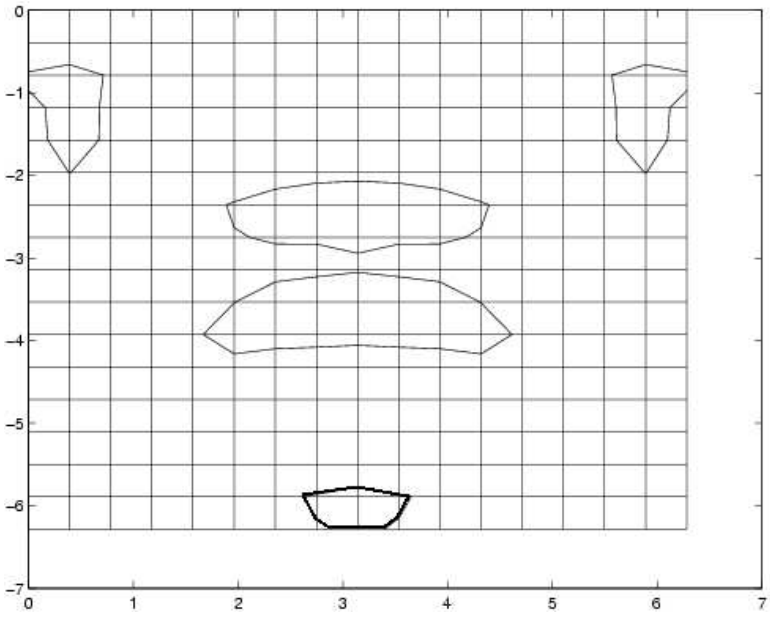


Fig. 11. Level-set violates problem construction. Here, the violation occurs on bottom boundary as highlighted.

Specifically, the Dirichlet-Neumann maps are not correct. There is nothing in the formulation of the level-set that dictates how the interface will evolve. That is its advantage over parameter methods. But the currently defined minimization scheme does not constrain the index profile in $L_\infty(\Omega)$. Hence, we must impose a barrier on where the level-set is allowed to roam. A question arises here as to what makes a

naturally imposed barrier. One simple approach is to perform basic truncation. That is, as the level-set approaches the boundary, we cut it off. However, in the evolution step, the update scheme has no knowledge of this truncation. It will continue to grow according to its minimization step. Since the level-set is unnaturally altered, the associated surface function will not be updated appropriately according to the modified gradient, $\delta\phi = -G|\nabla\phi|$. To state differently, the update step for ϕ_{n+1} will not guarantee a descent direction, because $\nabla\phi$ does not correspond with the new truncated a. Another potential hazard is that the level-set will continue to build at the truncation boundary and not model the problem accurately. But numerical experiments have shown that the former is the case. Thus, a better approach for constraining the domain is sought. A continuous approach is more desirable in this instance. One fix is to place it in as a penalty in the cost function. As with many penalty methods, the level-set minimization and the imposed constraint may be at odds with each other. We will view the level-set boundary interfaces analogously with what is commonly referred to as a potential barrier. In many physical situations, the potential energy of the system grows as you approach a boundary. A barrier in our situation can be thought of as a penalty function that increases indefinitely as you approach the boundary: $B(x) \to \infty$, as $x \to \partial\Omega$, [31]. We make the following definitions:

Definition 7.1

$$\Psi(x) = \begin{cases} \infty & x \in \partial\Omega \\ 0 & otherwise \end{cases} \tag{7.7}$$

Definition 7.2

Let $S_a \equiv supp(a - a_1)$. *Let* $\tilde{f}(a) \equiv \sup_{(x,y)\in S_a} |y|$. *Let a cell whose top boundary is adjacent to the top interface be denoted* Υ, *and further let it be described as* $A \times [y_1, y_2]$. *Finally, let* $\Upsilon_a \equiv A \times [y_1, \tilde{f}(a)]$. *We shall refer to this set as the* ***uniform set****.*

$\tilde{f}(a)$ is interpreted as the distance the level-set is from the interface. Figure 12 illustrates the set Υ_a.

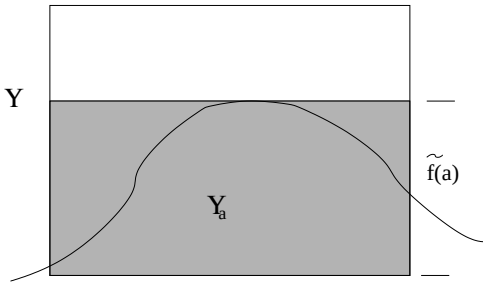


Fig. 12. Region Γ_k fill in.

In constructing a barrier function B certain conditions are required. First, as a sequence of functions, a parameter γ_n is chosen such that $B_{\gamma_n}(x) \to \Psi(x)$, as $\gamma_n \to 0$. As an example, consider $B_\gamma(x) = \frac{\gamma}{x+\gamma^2}$. Second, differentiability is necessary for use in the update scheme. The following barrier function will be used:

Definition 7.3 (Barrier Function)

Let

$$B_{\gamma_n}(a) = \gamma_n \sum_{\Upsilon_a} \int_{\Upsilon_a} (a - a_1)^2 P(x_2) dx_1 dx_2,$$

where $P(x_2)$ *has the following properties:*

$P \in C(0, -d)$, *and* $P(x_2) \geq 0$, *and* $P(x_2) \to \infty$, *as* $x_2 \to 0 \quad$ *and* $x_2 \to -d$.

First, notice that P is a function of the stratified distance only. Second, we state the gradient of this function, for it will be used in the update scheme. A straightforward calculation yields the gradient

$$\langle G(a), \delta a \rangle = \int_{\Omega} 2(a - a_1) P(x_2) \delta a.$$

The implementation of the above barrier method is as follows: Find $a \in \mathcal{A}$ such that

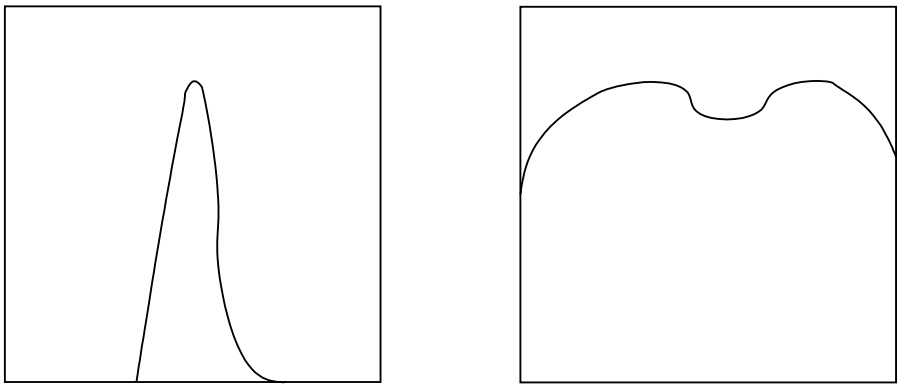
$$\min_{a \in \mathcal{A}} J(a) + B_c(a)$$

is found.

This definition of the barrier function is numerically infeasible, with singularities at the boundary. To fix this, we first consider $P \mid_{\partial\Omega} = \mathcal{P}$ for some fixed large $\mathcal{P}$. The $\mathcal{P}$ will be adjusted higher as level-sets continue to approach the boundary. Further, since the goal is to keep the level-set from reaching the boundary, the primary property from $a(x_1, x_2)$ is its stratified component. The distance of S_a is from the interfaces is important. Thus, a sharp thin profile that approaches a boundary should equally be penalized as much as a thick profile. See Figure 13. However, the implementation of such a function has proven to be very difficult. Indeed, a barrier function that better represents this situation is

$$\tilde{B}_{\gamma_n}(a) = \gamma_n \int_{\Upsilon} (a - a_1)^2 \chi_{\Upsilon_a} P(x_2) d\Upsilon. \tag{7.8}$$

This implements a uniform mass (the uniform set) across rectangular cells where its volume is proportional to the maximum height of the level-set. See Figure 12. However, (7.8) is not differentiable. Even though B_{γ_n} from Definition 7.3 does not penalize thin profiles well, it seems to work adequately in practice. A conceivable problem could be that the level-set forms a cusp. This cusp can be avoided by re-



(a) (b)

Fig. 13. (a) A thin profile. (b) A thick profile. Each have the same height, and should have the same weight. But this is inconsistent with the barrier function.

stricting to level-sets with Lipschitz boundary, which is true for a C^2 surface profile. This does not necessarily hold for a piecewise defined surface profile in the discretization steps. However, in numerical experiments, the cusp shapes have not presented a problem. Indeed, in practice, when a shape profile approaches the boundary, the best strategy has been to restart with a different profile. Usually, initial profiles that stay sufficiently far from the boundary pose no problems during the evolution step. Examples of this will be shown in Chapter VIII.

E. Appropriate Update Scheme of the Level-Set Surface Function

Due to the nodal values being at the corner of several planes (maximum of 6) of the surface profile function, ϕ, the gradient is not well defined at the nodes. See Figure 14. Therefore, the appropriate update (see (6.3)) needs the gradient to be averaged according to the local gradients. We will implement a finite difference scheme by

choosing an appropriate average to minimize the error. Initial attempts employed a basic finite difference across the mesh, but poor convergence was observed, and iterations failed to continue in a descent direction after the error reached a certain minimum, falsely flagging a local minimum. Enhancements to the averaging scheme have yielded much better results obtaining minimums of the cost functional within specified tolerances. The best scheme found thus far has been to compute alternate finite differences across 7 points. This gradient's norm is computed for use in the update function. Referring to Figure 14, the 7 points in the figure comprise 6 planes. The symmetry is altered between the left half and right half of the domain space. Thus, a corresponding picture (not shown) exists with diagonals from NE to SW.

We define the gradient at a node p^j from the six faces as follows: Letting p^j be the specific node, we define p_i^j to be the appropriate node on face i relative to node p^j, as defined in Figure 14. Let $\widehat{G}_{j,i} = \langle \Delta x_{j,i}, \Delta y_{j,i} \rangle$, where the differences Δx and Δy are appropriately defined by the points $\{p^j, p_i^j, p_{i+1}^j\}$ comprising the planar face i. I.e., $\Delta x_{j,1} = \langle \frac{\phi(p_1^j) - \phi(p^j)}{h_x}, \frac{\phi(p_2^j) - \phi(p^j)}{h_y} \rangle$. Given $\widehat{G}_{i,j}$, $i = 1..6$, let

$$G(p^j) \equiv |\sum_{i=1}^{6} \frac{1}{6} \widehat{G}_{i,j}|.$$

From its definition, its not clear how well this approximates the gradient of a C^1 function. The standard central differences uses a 5 point scheme, whereas a 7-point scheme is used here. The following lemma yields its order of approximation.

Lemma 7.4

The 7-point gradient scheme is a second order approximation for $\phi \in C^2$

Proof. Using stencil notation, we consider the matrix of coefficients used in the finite differences: Denote by S_x the stencil in x-direction, and S_y the stencil in y-direction (see Figure 15).

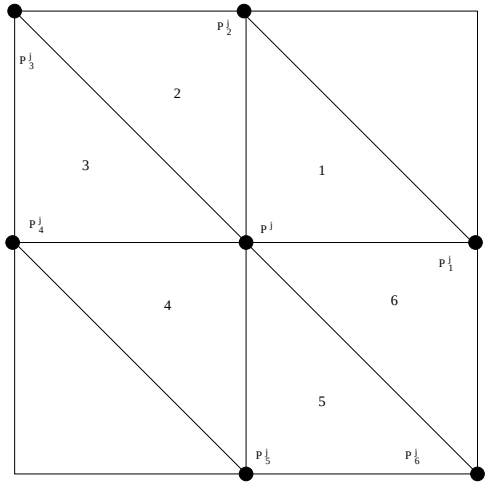


Fig. 14. 6 faces for an internal node.

$$\underbrace{\frac{1}{6h}\begin{bmatrix} -1 & 1 & 0 \\ -2 & 0 & 2 \\ 0 & -1 & 1 \end{bmatrix}}_{S_x} \qquad \underbrace{\frac{1}{6h}\begin{bmatrix} 1 & 2 & 0 \\ -1 & 0 & 1 \\ 0 & -2 & -1 \end{bmatrix}}_{S_y}$$

Fig. 15. X stencil and Y stencil.

Consider the following nodal point map in Figure 16.

Using a Taylor series approximation centered at $(0,0) \to (x,y)$, applied to each of the 6 nodal points, we associate the value $\phi_{i,j}$, as depicted in Figure 16 where $-1 \leq i,j \leq 1$ with the corresponding Taylor series expansion. Then the map

$$\sum_{i,j} \left(S_x(i,j) + S_y(i,j)\right) \phi_{i,j}$$

yields

$$\phi_x(x,y) + \phi_y(x,y) + h^2(\phi_{x,x,x} + \phi_{y,y,y}) + O(h^3).$$

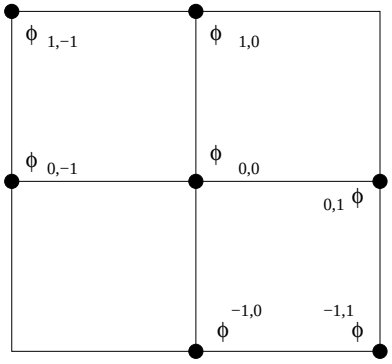


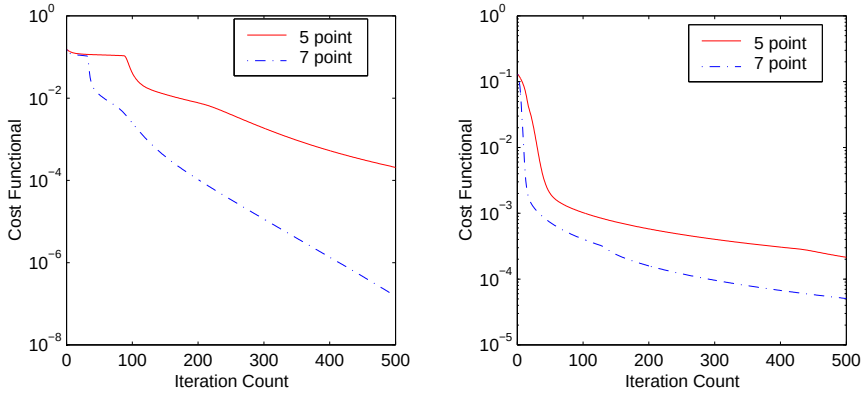
Fig. 16. Nodal points on the stencil grid.

Stated differently, the gradient scheme applied to the Taylor series yields a second order approximation.

□

If the surface profile ϕ has sufficient smoothness, then the 7-point gradient is adequate for approximation at the nodal points. By comparison, the symmetric 5-point stencil for center, north, south, east, and west nodes yields also a second order approximation. Surprisingly, the two approximations match each other out to h^5. That is, $|D_4 - D_6| \approx O(h^6)$. In fact, the 5-point has a smaller error coefficient on h^6. This leaves open the question as to why the 6-point scheme is a better approximation. One possible reason could be that an approximation to the smooth level-set may not accurately reflect the piecewise planar functions. An average around the local planar faces that contribute to the nodal point appears to be a better approximation for the update scheme, than a standard 5-point stencil. Also investigated was a 9-point stencil where all nodes at the far corners we used, but poor convergence was observed as well as compared with the 7-point. Shown in Figure 17 is a pair of plots showing the convergence results between the two schemes. The two plots represent changed

parameters. (a) is with profile index 1 and 2.8. (b) is with a profile index 1, 2.5. The tolerance in the error for the cost functional is on the order of 10^{-4}.



(a) (b)

Fig. 17. A comparison of the gradient schemes for 5 points and 7 points. Each plot compares iteration counts versus the error in the cost function using one gradient scheme versus the other. Notice the 7 point yields slightly better results.

F. Multi-Layered Operator

In the solve for the gradient of this system, we require two solves, one for the direct forward problem, and another for the adjoint. In the multi-layered structure, the computational cost for a direct forward solve is given by $(MNL)^3$, where M is rows, N is columns, and L is layers of the mesh structure. This can be improved by taking advantage of the repeat in the layers. This multi-layered structure lends itself very appropriately to Domain Decomposition methods [32]. We will describe the multi-layer structures within this framework of Domain Decomposition. First, we describe

briefly the outline of a Domain Decomposition method. We refer the interested reader to [32, 13] for details. Given a set Ω, and a system to solve for (a PDE plus BC's), one breaks the region into 2 or more smaller domains, $\Omega_1, \Omega_2, \ldots, \Omega_n$. The solution U on the smaller regions each satisfy a similar PDE with appropriate BC's. We denote the solution on the smaller regions by U_j. How the regions are split depends on the type of problem and geometry of the original domain. The regions Ω_j may overlap each other, in which case the methods are referred as Overlapping Domain Decomposition. In our problem we will consider **non-overlapping** domains. Choosing the correct boundary conditions are vital to the Domain Decomposition method, for they define how the separate subproblems interact with each other. For example, forcing continuity of the solution across the boundary is essential in most problems where Domain Decomposition is used. This defines Dirichlet conditions. In second order problems, more information is required, and Neumann conditions are used to make the problem well posed. Other BC's can be used, as in Neumann-Neumann. The essential idea in a Domain Decomposition method is to reduce computational cost by solving a set of smaller subproblems. In an iterative Domain Decomposition strategy the solution in one domain supplies the boundary data in the next, and then the solution in that domain supplies the first, and so the process is repeated until the individual solutions converge to the original full problem. This general procedure is called an **iterative substructuring method**. We describe a generic Dirichlet-Neumann iterative method here [32]: A domain is split into two or more regions, and a black and white coloring scheme is employed on the regions. Let I_B be the set of Ω_i that are colored black, and I_W the remainder; let $\Gamma_{i,j}$ represent the boundary, $\overline{\Omega_j} \bigcap \overline{\Omega_i}$. One solves for the following pair of iterative PDE's (general notation is used here, for no attempt is made at analyzing the procedure). Let U_i^k solve the following

generalized PDE with BC's:

$$\text{(7.9)} \qquad \begin{cases} LU_i^{k+1} &= f & \text{in } \Omega_i,\ \forall i \in I_B, \\ \Phi(U_i^{k+1}) &= \theta\Phi(U_j^k) + (1-\theta)\Phi(U_i^k) \quad \forall j \in I_W, \ \text{ on } \Gamma_{i,j} \neq \emptyset, \end{cases}$$

where θ ($0 \leq \theta \leq 1$) is a relaxation parameter, and Φ is some compatibility function, like $\Phi(\nu) = \nu$. The paired problem is:

$$\text{(7.10)} \qquad \begin{cases} LU_j^{k+1} &= f & \text{in } \Omega_j,\ \forall j \in I_W, \\ \Psi(U_j^{k+1}) &= \Psi(U_i^k) \quad \forall i \in I_B, \ \text{ on } \Gamma_{i,j} \neq \emptyset. \end{cases}$$

Here, Ψ is another compatibility function, as, for example, enforcing Neumann conditions, $\Psi = \frac{\partial}{\partial \nu}$.

This framework defines the iterative procedure where information from the solution of one or more sub-domains supplies the boundary data into another subproblem, and the process continues. In our problem, we have attempted to do just that. But, instead of splitting the Dirichlet and Neumann operators as described above, we are imposing them simultaneously. They are essentially coupled via the D-N operators. We view the problem this way to make the substructuring scheme identical for each layer. We present that in the following subsection.

1. Domain Decomposition Interface Conditions

Depicted in Figure 18 is a schematic of the multi-layers and the internal transmission and reflections between adjacent layers. The Helmholtz problem for each layer is identical except for the incoming waves from above and below.

We look at the following problem description. All interfaces Γ_j lie in a homogeneous region, and due to the repetition of the layered structure, all of the homogeneous regions are the same, with a squared refractive index of a_1. Thus, the associated

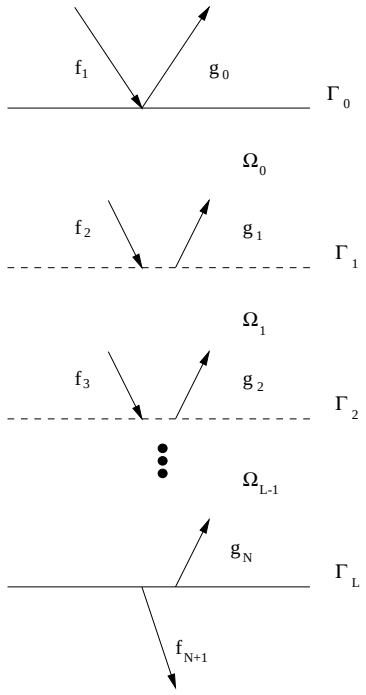


Fig. 18. Multi-layer with internal reflections and transmissions.

propagating mode coefficients β_n^j will be the same for each j, and so we drop the j notation for a simpler β_n. Similarly, we drop the notation on the $D-N$ operator for $Tf = \sum_n i\beta_n f^n e^{inx}$, where f is evaluated on interfaces Γ_j, and f^n are Fourier coefficients. Tf_j, and Tg_j are then well-defined. In Ω_j we have the solution U_j that solves the periodic Helmholtz equation, $j = 0, \ldots, L-1$,

$$(\Delta_\alpha + \omega^2 k_j^2) U_j = 0, \tag{7.11}$$

with the following boundary conditions:

$$TU_j - \frac{\partial U_j}{\partial \nu} = 2Tf_j, \quad on\, \Gamma_j, \tag{7.12}$$

$$TU_j - \frac{\partial U_j}{\partial \nu} = 2Tg_j, \quad on\, \Gamma_{j+1}. \tag{7.13}$$

Focusing on the interface Γ_j we force Dirichlet and Neumann conditions to match. This yields $U_j|_{\Gamma_j} = U_{j-1}|_{\Gamma_j}$ and $\frac{\partial U_j}{\partial \nu} = \frac{\partial U_{j-1}}{\partial \nu}$. So, substitution into (7.12) yields the following with a similar substitution into the bottom boundary condition on the subdomain Ω_{j-1}. First replace j with $j-1$ into (7.13). Coupling the two formulas and adding together, the following result is obtained:

$$TU_{j-1} = T(f_j + g_{j-1}).$$

Equating coefficients gives $f_j = U_{j-1}|_{\Gamma_j} - g_{j-1}$, one of the desired interface update formulas. Note that the Helmholtz equation (7.11) above indicates the dependence on the index profile k_j. But in light of Chapter V on multi-layered structures, we see that $k_0 = k_1 = \cdots = k_{L-1}$. A non-overlapping iterative method can be applied in this situation to solve the multi-layered structure.

2. Substructure Iteration Scheme

Problem j: find $U_j \in H^1(\Omega_j)$ such that

$$\begin{aligned} \triangle_\alpha U_j + \omega^2 k^2 U_j &= 0, && \text{in } \Omega_j, \\ -\frac{\partial U_j}{\partial \nu} + TU_j &= 2Tf_j, && \text{on } \Gamma_{j-1}, \\ -\frac{\partial U_j}{\partial \nu} + TU_j &= 2Tg_j, && \text{on } \Gamma_j. \end{aligned}$$

1. Set $f_j^0 = 0$, $j = 2, \ldots, L$, and $g_j^0 = 0$, $j = 1, \ldots, L-1$.
2. For $k = 1, \ldots,$ convergence,
   Solve problem 1 with $g_1 = g_1^{k-1}$, obtaining solution U_1^k.
   Set $f_2^k = U_1^k|_{\Gamma 1} - g_1^{k-1}$.
   For $j = 2, \ldots, L-1$,
      Solve problem j with $f_j = f_j^k$, $g_j = g_j^{k-1}$, obtaining solution U_j^k.
      Set $g_{j-1}^k = U_j^k|_{\Gamma_{j-1}} - f_j^k$, and $f_{j+1}^k = U_j^k|_{\Gamma_j} - g_j^{k-1}$.
   End
   Solve problem L with $f_L = f_L^k$, obtaining solution U_L^k.
   Set $g_{L-1}^k = U_L^k|_{\Gamma L} - f_L^k$.

```
End
```

In Chapter VIII, we discuss numerical tests of convergence of this algorithm and view several test runs. The algorithm above utilized the solution along the interfaces to supply information as a boundary condition for the next layer. Specifically, we look at the reflections and transmitted modes as the boundary conditions for the adjacent layers. This method simulates a propagating wave as it initially passes through the structure, by feeding each layer with the next stage of interference patterns as another layer is reached. The result develops into an iterative routine. In what follows, we view an alternate method to solving the multi-layered structure. Here we will view the problem in the same way, but this time consider that the internal reflection and transmissions are unknowns and solve them in a system.

3. Matrix Operator for Multi-layers

First, define a new operator. Each layer is independent of the other, and what constitutes the multi-layered structure is the interaction between each interface. Thus, each layer is viewed as a separate problem, as the solution in each layer satisfies (7.11). The incoming waves from above (the transmission from problem above) and below (the reflection from below) are the inputs. The outputs are the layer's respective outgoing waves across the interfaces. By way of illustration, Figure 18 shows the second layer with incoming wave from above being f_2, and from below as g_2. After scattering, the resulting reflected wave for the top is g_1, and the transmitted wave for below is f_3.

We view this input/output relation in operator form:

Definition 7.5 (Scattering Operator)

Let

$$\mathcal{L}_a : [\ \textit{incoming waves}\ ] \rightarrow [\ \textit{outgoing waves}\ ]$$

represent the scattering of light through the given profile a. The incoming waves are denoted by $\{f_i, g_i\}$ and the outgoing waves are denoted by $\{f_o, g_o\}$. Then

$$\mathcal{L}_a \begin{bmatrix} f_i & g_i \end{bmatrix}^T = \begin{bmatrix} g_o & f_o \end{bmatrix}^T$$

(Note the dependence on the index profile a.) Thus, in the example given above, the operator form becomes

$$\mathcal{L}_a \begin{pmatrix} f_2 \\ g_2 \end{pmatrix} = \begin{pmatrix} g_1 \\ f_3 \end{pmatrix}.$$

In general, the operator applied to Ω_i yields

$$\mathcal{L}_a \begin{pmatrix} f_{i+1} \\ g_{i+1} \end{pmatrix} = \begin{pmatrix} g_i \\ f_{i+2} \end{pmatrix}.$$

In the computational form of this operator, we view it as a $2x2$ block form,

$$\mathcal{L}_a = A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}. \tag{7.14}$$

The computation of the wave solution of the full L-layered system is obtained by solving for the unknown incoming and outgoing waves at the interfaces. After each such solution is obtained (different methods are employed here for comparison), each layer can be solved separately using $\mathcal{L}_a$. To obtain the final reflection and transmission coefficients, the full solution is not necessary under this scheme. Under this sub-block scheme, the values at the boundaries are obtained. Since the forward map F, (3.9), is the vector of propagating modes, computed via the solution on the top and bottom interfaces, it follows that the solution does not need to be computed on the entire domain. However, the gradient descent method for the level-set routines do require the gradient on the entire finite element grid, and thus, an efficient solver on the

multi-layered is desired. To illustrate the sub-block system, we analyze the following 4-layer block tridiagonal matrix:

$$\begin{bmatrix} -I & A_{12} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & A_{22} & -I & 0 & 0 & 0 & 0 & 0 \\ 0 & -I & B_{11} & B_{12} & 0 & 0 & 0 & 0 \\ 0 & 0 & B_{21} & B_{22} & -I & 0 & 0 & 0 \\ 0 & 0 & 0 & -I & C_{11} & C_{12} & 0 & 0 \\ 0 & 0 & 0 & 0 & C_{21} & C_{22} & -I & 0 \\ 0 & 0 & 0 & 0 & 0 & -I & D_{11} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & D_{21} & -I \end{bmatrix} \begin{bmatrix} g_0 \\ g_1 \\ f_2 \\ g_2 \\ f_3 \\ g_3 \\ f_4 \\ f_5 \end{bmatrix} = \begin{bmatrix} -(A_{11})f_1 \\ -A_{21}f_1 \\ 0 \\ 0 \\ 0 \\ 0 \\ -D_{12}g_4 \\ -(D_{22})g_4 \end{bmatrix}.$$

The sub-blocks, $A, B, C,$ and$D$ are denoted for each layer, but they are all equivalent. We can view this matrix more efficiently with the following mapping sequence substitution for the blocks: Let the sequence $[0, 1, 2, 3, 4, 5]$ map to the following sub-block matrices:

0	$A_{1,1}$	$A_{1,2}$	$A_{2,1}$	$A_{2,2}$	$-I$
0	1	2	3	4	5

Then, the tridiagonal block can be viewed as

$$\begin{pmatrix} 5 & 2 & & & & & & \\ 0 & 4 & 5 & & & & & \\ & 5 & 1 & 2 & & & & \\ & & 3 & 4 & 5 & & & \\ & & & 5 & 1 & 2 & & \\ & & & & 3 & 4 & 5 & \\ & & & & & 5 & 1 & 0 \\ & & & & & & 3 & 5 \end{pmatrix}. \tag{7.15}$$

We employ a basic tridiagonal solve on this system, and observe the computational effort is a function of the cost to compute the 2×2 block matrix of $\mathcal{L}_a$. Once

that is computed, all matrices in the system are computed using a symbolic tridiagonal solve, using block matrix solves based on the above map. The dimensions of the block matrix A is then $2N \times 2N$, where N is the horizontal mesh size. The computational effort to compute this system is a basic Gaussian elimination step on the direct forward problem applied $2N$ times, totaling $O(2N[N(M+1)]^3)$. If we add to this the $2L$-block tridiagonal system on the order $O((4L))$, multiplied by the Gaussian elimination of each $N \times N$ block (order $O(N^3)$), then we have $O(2N^4(M+1)^3 + 4LN^3)$. A full direct Gaussian elimination on the L layered system has $O(LN(M+1))^3$. If $L = N = M$, then we have N^7 versus N^9. For large L, it is clear that the block tridiagonal will be superior. Thus, the multi-layered structure, for large L, has an order L^2 speed up over a direct method, using a basic tridiagonal solve.

For an iterative procedure like GMRES, we found the computational steps involved were too inefficient. Often, GMRES computed too many iteration counts to be dependable in the level-set routines. As the problem domain changed (evolving level-sets), the system varied enough to exhaust the iteration count before adequate convergence could be obtained. We describe the basic framework for an iterative routine. The action operator in the iteration routines is computed via a sweep through the layers. Since the layers are all the same, there is one direct forward problem that is used for all the layers. The framework involves computing the block system, (7.14). To compute the action on the block tridiagonal system, we perform two forward problem solves per layer, then we sweep through the layers by applying Definition 7.5 by feeding the next layer the wave solution from the previous. This strategy requires $2L$ solves from the $N(M+1)$ sparse system, which comes to a computational cost of $O(2L(N(M+1))^3)$. Now, if the systems were comparable for each iteration in the level-set evolution scheme, then we could provide further analysis. But what has been observed is by changing the level-set slightly, the problem's max iteration count

is reached frequently, before adequate convergence occurs. This approach then has been postponed until further analysis can be performed.

CHAPTER VIII

NUMERICAL RESULTS AND CONCLUSION

In this chapter we present some results on the various optimal designs that are possible within the framework of multi-layered structures. In the design strategy we have tried to keep simple structures. We have tried to do this by starting with simple initial structures and see how the level set evolves under 1, 2, 3, 4, 5, and more layers.

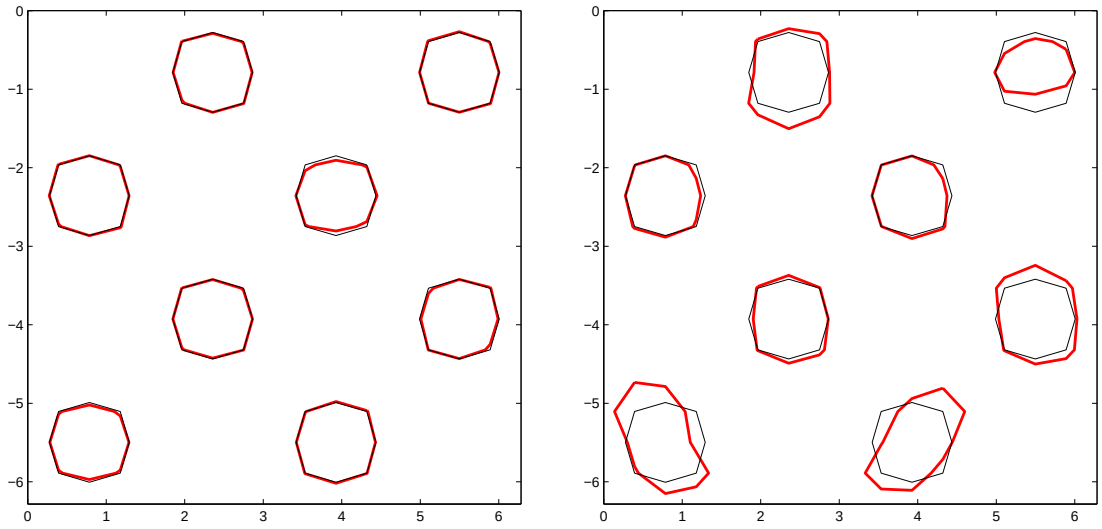
The following list of questions will be considered in this chapter:

1. Does the complexity of an evolved structure increase or decrease with multi-layers?
2. What do complicated initial structures do under level-set evolution?
3. Is the sensitivity of a multi-layered structure dependent on the initial profile?

We address these issues in order.

A. Miscellaneous Observations

From the previous sections on ill-posedness and the design problem, a few observations and notes are in order. First, it has been observed that starting with a complex initial profile, the iterated profile does not change so much as compared with starting with simpler structures. Thus, as the following two diagrams in Figure 19 indicate, they produce significantly different propagating modes, while not drastically altering their initial profiles. Of course, this suggests a high degree of sensitivity.



(a) (b)

Fig. 19. Comparison of high degree of sensitivity. Initial profile is seen in black, compared with the evolved profile shown in thick red. These profiles represent anti-reflective structures. Both have indices of refraction $k_1 = 1$, $k_2 = 2.9$. (a) $\omega = 1.9$, (b) $\omega = 1.3$.

We note that simpler initial profiles provide for larger deviation in the evolved shape. For example, Figure 20 shows how a simple shape evolves for non-reflective surfaces. Also, notice how (b) and (c) are similarly shaped, yet give differing evolved profiles.

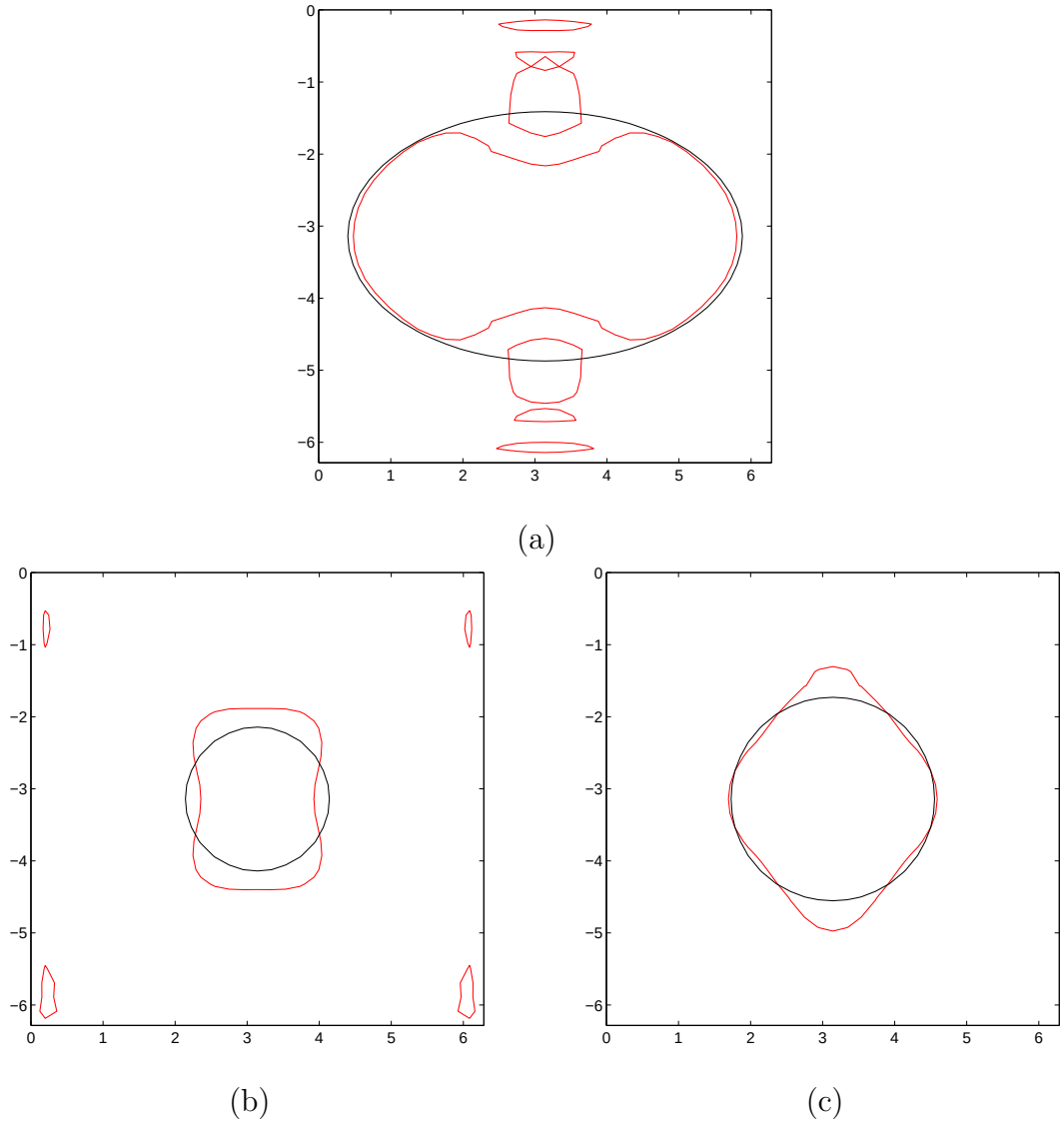


Fig. 20. Initial simpler structures evolve and give interesting, more complex shapes. Depicted here are simple shaped initial profiles evolving differently to yield the same propagating modes, which is $R = 0, T = 1$, error $\leq 10^{-7}$ in cost function. (a) Initial profile is a wide ellipse.(b) and (c) are circles with different radii.

B. Multi-Layered Structure Results

The following figures show the success of the multi-layered method. It also demonstrates the power of the level-set method, as profiles evolve into various interesting shapes. The symmetry noticed in the profiles comes from an incoming plane wave with zero incident angle. If the initial profile is symmetric, the evolved profile will remain symmetric as well.

In Figure 21, we show two profiles that converge to the same target reflection and transmission modes for a 3 layered system, but starting from different initial profiles. The final iterated profiles are shown in thick red contours, while the initial profiles are in thin black.

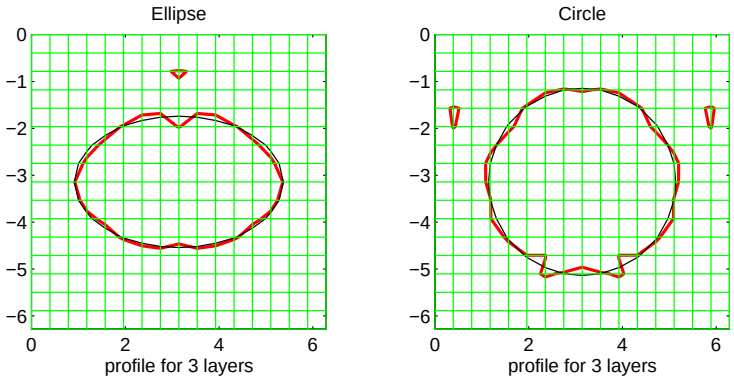


Fig. 21. 3 layered structure.

Figure 22 shows another 3 layered iteration shown from initial curve. Notice the holes in the final structure. The freedom of the profile to develop this kind of geometry is natural. For this plot we used material with $k_1 = 1$, $k_2 = 2.9$, $\omega = 1.9$. This profile represents a non-reflecting profile, with the target distribution given in the modes as $-1 \to .3, 0 \to .4, 1 \to .3$.

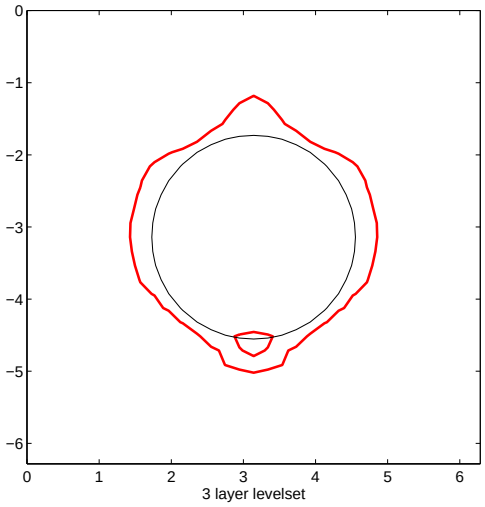


Fig. 22. Another 3 layered structure.

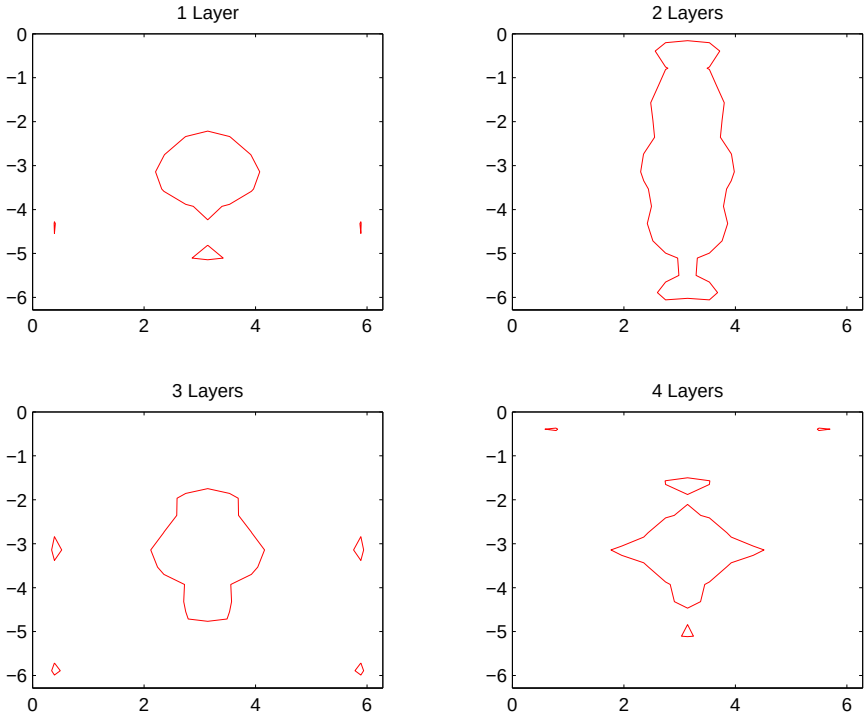


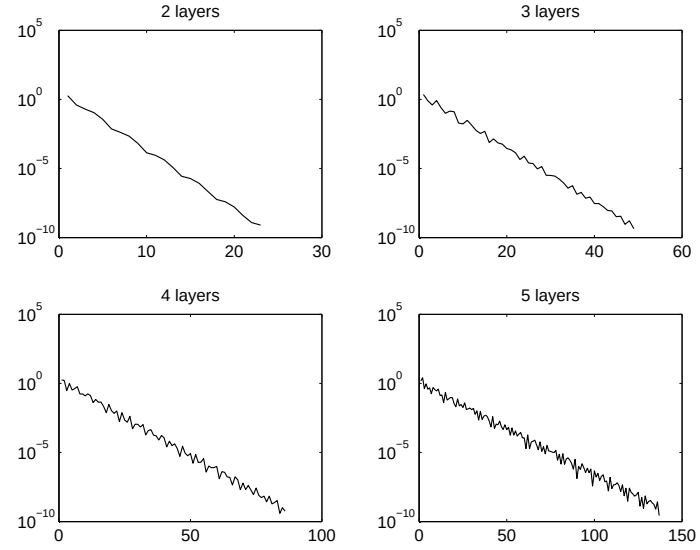
Fig. 23. No reflections with multi-layered structures.

In Figure 23, 4 examples are shown. They are successively increasing in layers. They each represent the same target reflections and transmissions within desired tolerances of 10^{-4}. The target reflection and transmission is no reflections and equal distribution on transmission.

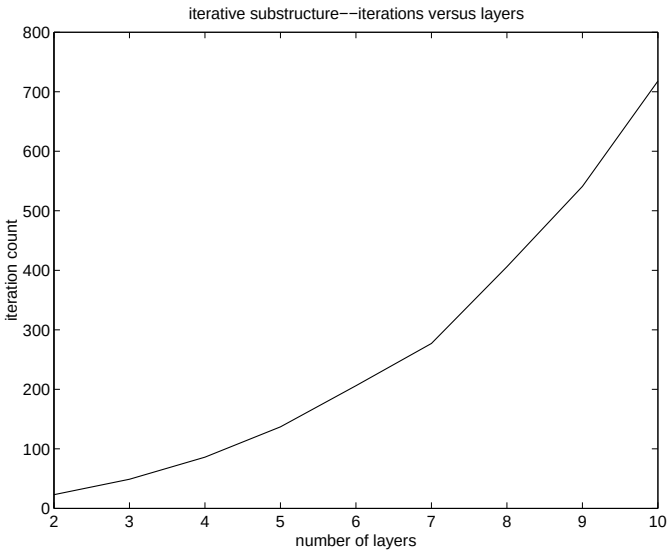
C. Iterative Substructuring

Shown in Figure 24(a) is the convergence of the iterative substructuring scheme applied to 2, 3, 4, and 5 layers. The plots show the iteration sweeps versus the error (l_2 error) of the solution, with the error on a log scale. The solution on the interfaces all converge to within 10^{-9}. Notice also that the iteration count increases quadratically with the number of layers, as shown in Figure 24 (b).

The convergence of this algorithm cannot be guaranteed, for there are cases where it fails. The failures are observed with increasing frequencies ω and increasing layers. Figure 25 shows the error versus iteration count of one such failure. The substructuring algorithm (page 80) simulates the initial scattering as a wave begins to traverse the layers. The idea is to settle on a converging scheme as the iterations accumulate more layers in the profile. However, a more robust scheme should be developed to account for the higher frequency divergence.



(a)



(b)

Fig. 24. (a) 4 plots show log of error versus iteration count in the iterated substructuring of 2, 3, 4, 5 layered structures. (b) Quadratic profile of iteration steps to desired tolerance as number of layers increase.

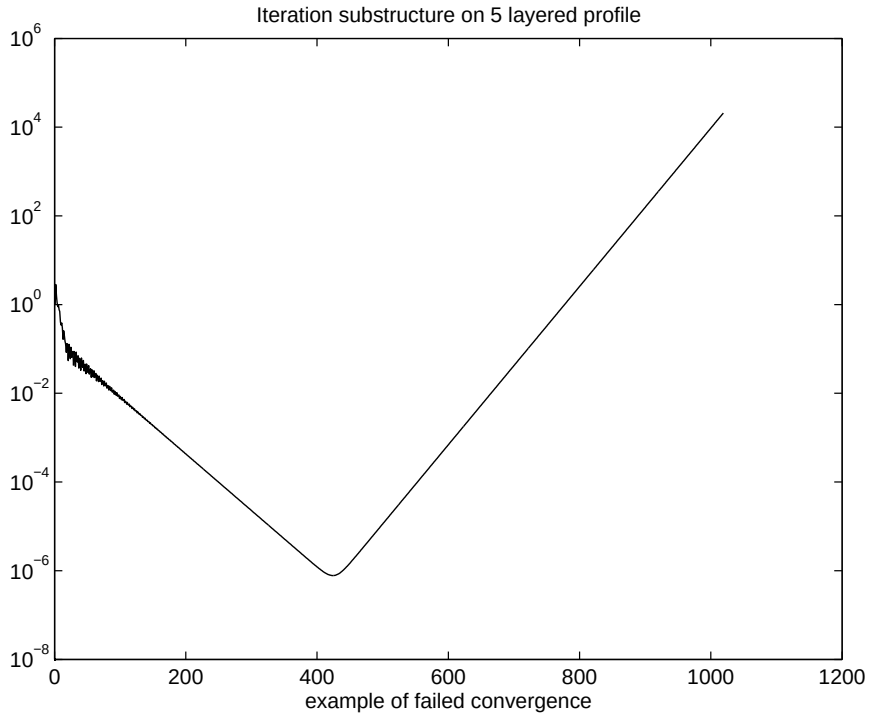


Fig. 25. Failure of convergence of iterative substructuring scheme.

D. Superprism Phenomena

In 1998, Kosaka et al., from NEC, Tohoku University, and NTT Opto-electronics laboratory [20], showed the existence of a special optical phenomenon termed a **superprism** effect. Essentially it is the ability to cause high dispersion of the propagating modes (transmission) with relatively small change in the incident angle. The structures used in their analysis were 2-D photonic crystals. In their research they were interested in the effect of photonic band gap structures, which prohibits the propagation of light of certain frequencies. In their observations they discovered the superprism effect as it relates to anomalous behaviors. The effect is observed near the band edge in photonic band gap structures. In this experiment, we try to recreate

the superprism effect. In the following figures, we are looking at transmitted modes versus the incident angle. Each different figure is from a specific frequency, in the range of $[.4, .5]$. We keep in mind that the frequency and incident angle determine the propagating modes. To keep the analysis simple, we consider small frequencies which keep the reflection and transmission to single zero-order modes.

We consider a circle profile on a 5 layered structure. The incident angle remains fixed at 0 radians, and the refractive index between the circular region and the surrounding medium is 2.9833 and 1, respectively. Depicted in Figure 26 is the transmission energy. Notice the transmission drops rapidly and stays significantly small for the range of frequencies between approximately .44 and .54, where a photonic band gap appears.

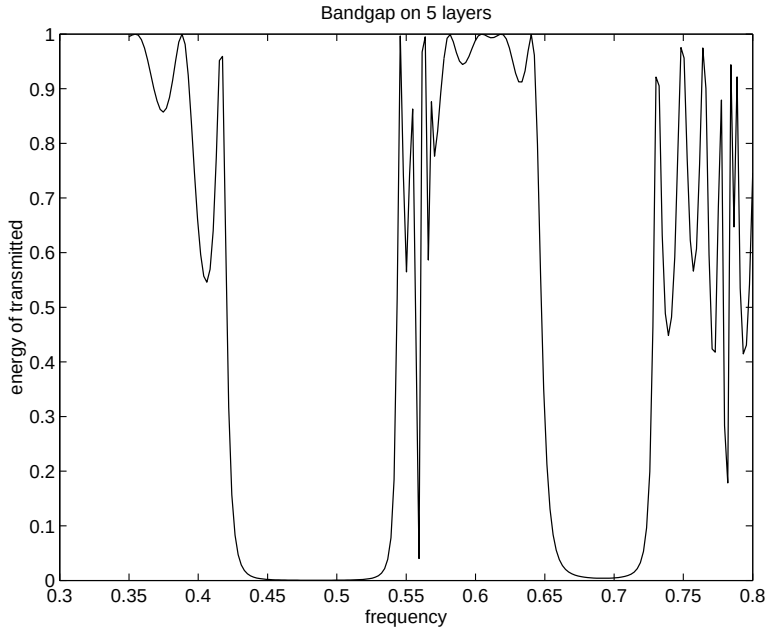


Fig. 26. Band gap shown for a 5-layered circular profile.

E. Future Work

The ongoing research for this work involves carefully analyzing the convergence of the level-set method. Its success in this work, using gradient step methods, shows promise for improvements, with better finite difference schemes to be developed. In this work, the finite element method employed a rectangular grid. Future work can include an adaptive mesh to trace the boundary of the level-set, since its local behavior decides where it should move in the next step. We recall that global properties of the surface profile (regions bounded away from the level-set) will also determine the evolution. Different surface profiles can be explored that can dramatically change the ultimate evolution of the level-set.

As the level-set evolves, it was observed that it is possible for certain cusps to form, thereby making the structure very difficult to fabricate. A penalty method on the level-set surface function could be employed, eg. $\|\phi\|^2_{H^k(\Omega)}$, for some $k \geq 1$, to help remedy this pitfall. Future design strategies should include this into the cost functional.

Finally, the optimization scheme fixed the incident angle and frequency for a given profile. By establishing faster multi-layered solvers, we can incorporate that into a robust, multi-angle, multi-frequency, optimization scheme. This will provide for a range of incident angles and a narrow band of light (varying frequencies) to scatter at the desired modes of reflection and transmission. Such problems are of great practical and theoretical interest and also provide more data so that solutions may not be as severely under-determined.

F. Conclusion

We have developed an iterative technique to compute optimal shape designs for layered periodic structures by use of a Level Set method. The structures evolve from simple initial profiles with varying degrees of increased complexity. A multi-layered solver was developed to help the optimization scheme.

REFERENCES

[1] Y. Achdou, *Optimization of photocell*, Optimal Control Appl. Methods, 12 (1991), pp. 221-246.

[2] R. A. Adams, *Sobolev Spaces*, Academic Press, New York, 1975.

[3] H. Ammari, N. Bereux, and E. Bonnetier, *Analysis of the radiation properties of a planar anatenna on a photonic crystal substrate*, Math. Methods App. Sc., 24 (2001), pp. 1021-1042.

[4] G. Bao, *Finite element approximation of time harmonic waves in periodic structures*, SIAM J. Numer. Anal., 32 (1995), pp.1155-1169.

[5] M. Beals, *Propagation and Interaction of Singularities in Nonlinear Hyperbolic Problems*, Birkhäuser, Boston, 1989.

[6] M. Born and E. Wolf, *Principles of Optics*, 6th Edition, Pergamon, Oxford, 1989.

[7] L. C. Botten, N. P. Nicorovici, A. A. Asatryan, R. C. McPhedran, C. M. Sterke, and P. A. Robinson, *Formulation for electromagnetic scattering and propagation through grating stacks of metallic and dielectric cylinders for photonic crystal calculations, Part I. Method*, J. Opt. Soc. Am. A, 17 (2000), pp. 2165-2176.

[8] S. Brenner and S. Ridgeway, *The Mathematical Theory of Finite Element Methods*, Springer-Verlag, New York, 1994.

[9] J. E. Dennis and R. Schnabel, *Numerical Methods for Unconstrained Optimization and Non-linear Equations*, Prentice-Hall, 1983.

[10] D. Dobson, *Optimal design of periodic antireflective structures for the Helmholtz equation*, Eur. J. Appl. Math., 4 (1993), pp.321-340.

[11] D. Dobson, *Optimal shape design of blazed diffraction gratings*, Appl Math Optim., 40 (1999) pp. 61-78.

[12] D. Dobson, *Controlled scattering of light waves: optimal design of diffractive optics*, Control Problems in Industry, Proceedings of the SIAM Symposium On Control Problems, San Diego, 1995.

[13] Q. Du, *Optimization based nonoverlapping domain decompositoin algorithms and their convergence*, SIAM J. Numer. Anal., 39 (2001), pp. 1056-1077.

[14] M. P. Eastham, *The Spectral Theory of Periodic Differential Equations*, Scottish Academic Press, Edinburgh, 1973.

[15] J. M. Elson and P. Tran, *Coupled-mode calculation with the R-matrix propagator for the dispersion of surface waves on a truncated photonic crystal*, Phys. Rev. B, 54 (1996), pp.1711-1715.

[16] M. Frank and M. Collischon, *Dielectric multilayer grating designs with maximum diffraction efficiencies*, Opt. Eng., 37 (1998), pp. 1696-1702.

[17] E. Hecht and A. Zajac, *Optics*, Addison-Wesley, Reading, Mass., 1974.

[18] J. D. Joannopoulos, R. D. Meade, and J. N. Winn, *Photonic Crystals–Molding the Flow of Light*, Princeton University Press, Princeton, New Jersey, 1995.

[19] R. V. Kohn and G. Strang, *Optimal design and relaxation of variational problems, I*, Commun. Pur. Appl. Math., 39 (1986), pp. 113-137.

[20] H. Kosaka, T. Kawashima, A. Tomita, M. Notomi, T. Tamamura ,T. Sato, and S. Kawakami, *Superprism phenomena in photonic crystals*, Phys. Rev. B, 58 (1998), pp. 10096-10099.

[21] P. Kuchment, *Floquet Theory for Partial Differential Equations*, Birkhäuser Verlag, Switzerland, 1993.

[22] M. H. Lim, T. E. Murphy, J. Ferrera, J. N. Damask, and H. I. Smith, *Fabrication techniques for grating-based optical devices*, J. Vac. Sci. Technol. B, 17 (1999), pp. 3208-3211.

[23] D. G. Luenberger, *Optimization by Vector Space Methods*, John Wiley & Sons, New York, 1969.

[24] R. März, *Integrated Optics: Design and Modeling*, Artech House, Norwood, Mass., 1995.

[25] D. Maystre, *Electromagnetic study of photonic band gaps*, Pure and Applied Optics: Euro. J. Opt. Soc. A, 3 (1994), pp. 975-993.

[26] C. Morawe, J. C. Peffen, O. Hignette, E. Ziegler, *Design and performance of graded multilayers*, International Society for Optical Engineering,Denver, Colorado, Proc. SPIE, 3773 (1999), pp. 90-99.

[27] S. J. Osher and F. Santosa, *Level set methods for optimization problems involving geometry and constraints I. Frequencies of a two-density inhomogeneous drum*, J. Comp. Phys., 171 (2001), pp. 272-288.

[28] M. Neviere and E. Popov, *Electromagnetic theory of gratings: review and potential applications*, Part of the SPIE Conference on Theory and Practice of

Surface-Relief Diffraction Gratings: Synchrotron and Other Applications, San Diego, 1998.

[29] J. B. Pendry and A. MacKinnon, *Calculation of Photon Dispersion Relations*, Phys. Rev. Lett., 69 (1992), pp. 2772-2775 .

[30] R. Petit, *Electromagnetic Theory of Gratings*, Topics in Current Physics, 22, Springer-Verlag, Berlin, 1980.

[31] E. Polak, *Computational Methods in Optimization, A Unified Approach*, Academic Press, New York, 1971.

[32] A. Quarteroni and A. Valli, *Domain Decomposition Methods for Partial Differential Equations*, Clarendon Press, Oxford, 1999.

[33] W. Rudin, *Principles of Mathematical Analysis*, 3rd Edition, McGraw-Hill, New-York, 1976.

[34] F. Santosa, *A level-set approach for inverse problems involving obstacles.* ESAIM: COCV, 1 (1996), pp.17-33.

[35] M. Schwartz, *Principles of Electrodynamics.* Dover Publications, Mineola, New York, 1987.

[36] J. R. Shewchuk, *Triangle: engineering a 2D quality mesh generator and Delaunay triangulator*, First Workshop on Applied Computational Geometry, Philadelphia, Pennsylvania, 1996, pp. 124-133.

[37] J. A. Sethian, *An Analysis of Flame Propagation*, PhD. Thesis, University of California, Berkeley, 1982.

[38] R. Malladi and J. A. Sethian, B. Vemuri, *Shape modeling with front propagation: a level set approach*, IEEE T. Pattern Anal., 17 (1995), pp. 158-175.

[39] J. A. Sethian, *Level Set Methods and Fast Marching Methods*, Cambridge University Press, 2nd ed., Cambridge, 1999.

[40] J. A. Sethian and A. Weigmann, *Structural boundary design via level set and immersed interface methods*, J. Comp. Phys., 163 (2000), pp. 489-528.

[41] B. W. Shore, M. D. Perry, J. A. Britten, R. D. Boyd, M. D. Feit, H. T. Nguyen, R. Chow, G. E. Loomis, and L. Li, *Design of high-efficiency dielectric reflection gratings*, J. Opt. Soc. Am. A, 14 (1997) pp. 1124-1136.

APPENDIX A

MATLAB CODE

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Iterative levelset, gradient step method, Implicit use of levelset
%
%Mike Flanagan
%Modified June 2002
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%



%Let F be vector we perform least-square minimization on  <F,F>
% F = [R-r0,T-t0],  R_i = abs(r_i)^2
%
% J = DF...  DF_i = DR_i * 2 * conj(r_i)
%

%initial data is set through domainsetup.m

domainsetup
initK = K;
% Boundary  Top
% b0 is vector of reflective modes
% y0 is position of top boundary interface
% N0 is index of reflective modes
% b1 is vector of transmission modes
% N1 is index of transmission modes
% y1 is position of bottom interface


%let initial profile function be
% f = - (x-pi)^2 - (y-pi)^2

[x,y] = meshgrid(0:2*pi/(N):2*pi,-2*pi:2*pi/(M):0);
%Pfull = - (x-pi).^2 - (y+pi).^2 + 1 ;
%Pfull = 1.2 - exp(.1*((x-pi).*(x-pi) + (y+pi).*(y+pi)));
P = Pfull(:,1:N);
Pstart = P; Pgood = P;
Pstore = reshape(P,1,N*(M+1));
size(P)

if (exist('Pbeginagain'))
    if (~isempty(Pbeginagain))
    P = Pbeginagain;
```

```
    Pgood = P;
    disp('starting P with Pbeginagain');
    disp('press any key to begin')
    pause
else
    record = []; step = 0;
end
else
 record = []; step = 0;
end



[levelind,levelset,firstflag, firstpoint, secondflag, secondpoint,...
 diagonalflag, diagonalpoint,diagonalflag2 diagonalpoint2] =...
                      newfulltriarea(P,M,N,k1,k2);
domainsolve_noK;

hx = 2*pi/N;
hy = (top-bottom)/M;
Jfunctional_barrier;
Jgood = J1;
tol = 1e-7;
maxit = 20;
scale = .5;
regular = 0;
flag = 1;
countfail = 0;
countsuccess=0;
lastscale = 5;
maxscale = 5;

while((J1-sum1)>1e-7)
J2 = J1;
J1
step = step + 1
if (rem(step,5) == 0)
 save Pgood
end

if (rem(step,1000) == 0)
    disp('pausing for you to look')
    pause
end
getrealgradient_noK;
%tempbarrier;


Pold = P;
% G is now defined.
% The routine above recomputes G with new U and W solved through
```

```
% the adjoint solve  with no K, but a levelset.
%G = real(func_avg(U,W,L,M,N));
%G = newgradient(U,W,L,M,N);
%K2 = mexcomputeintegral2(top,bottom,M,N,levelind,levelset,firstflag,...
%firstpoint,secondflag,secondpoint,diagonalflag,diagonalpoint,Pvec,a1,a2);
%K = mean(K2,1);


%Klayered = buildlayer(K,L);
%size(G)


%flag = 0;
% FORCE GRADIENT
%if (flag == 1)
%    gn = pcg('JtranJx2_noK',-G,tol,maxit,[],[],[],top,bottom,bottomlayered,...
%    omega,alph,Klayered,f,g,Mlayered,M,N,SP,U,b0,b1,y0,y1,...
%    N0,N1,E0,E1,R,T,L,targetreflect,targettrans,FOURIER,regular,P,a1,a2,hx,hy);%
%        gn = -G;
%    end
%A = hx*hy*J'*J+diag(ones(1,N*M)*.001);

%x = pcg(A,G,tol,maxit,[],[],[]);

%gn = -G;


% if (norm(gn)<1e-13)
%     gn
%   'possibly bad PCG computation'
%   pause
%end

%[levelind,levelset,firstflag, firstpoint, secondflag,...
% secondpoint, diagonalflag, diagonalpoint] = newfulltriarea(P,M,N,a1,a2);

computephigrad

%U = reshape(U,N,(M+1)).';
%W = reshape(W,N,(M+1)).';
UWc = real(multilayergrad_pointwise(U,W,M,N,L));
UWc_vec = reshape((UWc.'),N*(M+1),1);

flag=0;
if (flag == 1)
    gn = pcg('JtranJx2_noK',UWc_vec,tol,maxit,[],[],[],top,bottom,...
            bottomlayered,...
            omega,alph,Klayered,f,g,Mlayered,M,N,SP,U,b0,b1,y0,y1,...
            N0,N1,E0,E1,R,T,L,targetreflect,targettrans,FOURIER,...
            regular,P,a1,a2,hx,hy);
    else
```

```
        gn = UWc_vec;
    end


gn = (reshape(gn,N,M+1)).';



P = Pold -scale*(real(gn)/(a2-a1) + gradbarrier).*absgrd;

domainsolve_noK
Jfunctional_barrier;
if J1>=J2
    flag = 0;
    Pbad = P;
    Jbad = J1;
    countfail = countfail + 1;
    countsuccess = 0;
else
    countfail = 0;
    countsuccess = countsuccess+1;
    Pgood = P;
    Jgood=J1;
    flag = 1;
    regular = 0;
    if countsuccess > 3
        lastscale = lastscale*2;
    end
    if (lastscale > maxscale)
        lastscale = maxscale;
    end
end
record = [record;[J1,flag,scale,sum1]];
if (flag==0)
    if (countfail>5)
        scale = lastscale*2;
    else
        scale = lastscale/2;
     end
    while(J1>J2 & scale>1e-8)

    scale = scale/2
   lastscale = scale;

%getrealgradient_noK
%UWc = multilayergrad_pointwise(U,W,M,N,L);
%computephigrad

 P = Pgood -scale*(real(UWc)/(a2-a1) + gradbarrier).*absgrd;

   domainsolve_noK
```

```
   Jfunctional_barrier;

   end

  if (J1>=J2)
   % current direction is bad direction.

    regular = 0;
    P = Pbad;
    J1 = Jbad;
    flag = 0;
    disp('Cant decrease anyfurther')
    pause
  end
  if (J1<J2)
    record = [record;[J1 2 scale sum1]]
    Pgood = P;
    Jgood = J1;
    regular = 0;
    maxit = 20;
    flag = 1;
  end


end
scale = lastscale;
if (flag == 1)
  Pstore = [Pstore;reshape(P,1,N*(M+1))];
  end
end
'Complete'
J1
```

```
function [levelindinv,levelset,firstflag, first, secondflag, second,...
 diagflag1, diag1, diagflag2, diag2] = newfulltriarea(Porig,M,N,a1,a2)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
% Level-Set configurator.
%
% Mike Flanagan
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


%[levelset,firstflag, first, secondflag, second, diagflag, diag]
%= newfulltriarea(Porig,M,N,a1,a2)
% Matlab function to generate the appropriate x,y coordinates for
% the levelset on the rectangular grid.
% levelset returns a flag indicating if the rectangle is below,
% above, or if there is an intersection point.
% given respectively by a1, a2, -1
% firstflag determines if the first intersection point is top,left,right
% secondflag determines if the second intersection point is bottom,right,left.
% The flags are 1,2,3,4  for top,left,bottom,right.
% diagflag is 1 for a diagonal, zero for not.
% Its t parameter is given in terms of
% Mike Flanagan, 2002
% inputs  Porig for levelset space surface profile
% M for rows
% N for cols
% Porig is (M+1 x N) matrix
% levelindinv is the reverse index into firstflag,secondflag,etc..
% from the rectangle element.
% I.e.,  levelindinv(5) is either an index to a levelset or -1 for no levelset.
% if it is an index it points to the cooresponding element in first,second

Psw = Porig([1:M],:);
% That means we look at P without one of the boundaries. Its MxN
Pe = Porig(:,[2:N,1]);
Pne = Pe([2:M+1],:);
Pse = Pe([1:M],:);
Pnw = Porig([2:M+1],:);

% determine if levelset passes through or not
% nw x se     ne x sw



flag1 = (Pnw.*Pse < 0 | Pne.*Psw < 0);
flagabove = ~flag1 & (Pne >0 | Psw > 0);

%flag is the MxN matrix of rectangles that have a levelset
```

```
% 0 is levelset.  1 is not a levelset
%(all one refractive index or another)
% for the above, its all refractive index associated with above the
% plane.



[I1,J1] = find(flagabove);

flagbelow = ~flag1 & (Pne <0 | Psw < 0);
[I2,J2] = find(flagbelow);

% flagbelow+flagabove  yields a truth table of
% all indicies of rectangle above or below
% a2 is above
% a1 is below


%subplot(3,1,1)
%imagesc(flagabove)
%subplot(3,1,2)
%imagesc(flagbelow)
%flag(I1,J1) = ones(length(I1),length(J1))*a2;

% to test to see if I1,J1  set  and I2,J2 set compliment each other...




%subplot(3,1,3)
%imagesc(flag1)



%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Next, we construct a vector for the appropriate cells to determine
% which ones are on the levelset and which ones are not.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%[m,n] = size(K);

levelset = zeros(M*N,1);
aboveind = (J1-1)*M+I1;
belowind = (J2-1)*M+I2;
levelset(aboveind) = ones(size(aboveind))*a2;
levelset(belowind) = ones(size(belowind))*a1;

%next we traverse the level set through the zeros, and analyze them
%separate.
levelind = find(~levelset);
% to convert to a column vector, we note that matlab is column major
```

```
% the element #E = (row,col)  ==== M*(col-1)+row
levelset(levelind) = ones(size(levelind))*-1;
%iind = floor((levelind-1)/N)+1;
%jind = rem((levelind-1)/N)+1;
nw = reshape(Pnw,M*N,1);
ne = reshape(Pne,M*N,1);
sw = reshape(Psw,M*N,1);
se = reshape(Pse,M*N,1);

% for ease of notation, we restrict to the levelset indicies
nw = nw(levelind);
ne = ne(levelind);
sw = sw(levelind);
se = se(levelind);
%find first point
%first look top,left, right

% pattern is to look top, left , then right
% top is given by success to t*(ne-nw) + nw, t >=0 and t <= 1
% bottom is given by success to similar but for sw,se
% left is nw,sw   right is ne,se
% diagonal is nw,se

%top
tt = -nw./(ne-nw);
TOP = (tt>=0 & tt<=1);
%left
tl = -sw./(nw-sw);
LEFT = (tl>=0 & tl<=1);
%right
tr = -se./(ne-se);
RIGHT = (tr>=0 & tr<=1);

%Logic tricks.....
%TOP is set of indicies with interesection on Top.
%LEFT is set of indicies with intersection on Left.
% If an index had both TOP and LEFT, then first intersection is T,
% second is L.  We construct First point by using negation of sets.
%First = TOP  interesection  ~TOP + LEFT   intersection  ~TOP + ~LEFT + RIGHT
%First = find( TOP | (~TOP&LEFT) | (~TOP&~LEFT&RIGHT));
%to identify the top,left,right, we associate
T = TOP;
TL = (~TOP&LEFT);
TR = (~TOP&~LEFT&RIGHT);

%check to see that the T,L,and R are seperate indicies
if (max(T+TL+TR)>1 | min(T+TL+TR)<1)
    display('Problem, T,L,R are wrong')
    pause
end
firstflag = T+2*TL+4*TR;
```

```
first = T.*tt+TL.*tl+TR.*tr;



%bottom
tb = -sw./(se-sw);
BOTTOM = (tb>=0 & tb<=1);

%To find second point, we look at Bottom, right and then left
%Second = find( BOTTOM | (~BOTTOM&RIGHT) | (~BOTTOM&~RIGHT&LEFT));
B = BOTTOM;
BR=~BOTTOM&RIGHT;
BL=~BOTTOM&~RIGHT&LEFT;
secondflag = 3*B+4*BR+2*BL;
second=B.*tb+BR.*tr+BL.*tl;

%diagonal nw-->se
td = -nw./(se-nw);
DIAG1 = (td>=0 & td<=1 & isfinite(td));
diagflag1 = DIAG1;
diag1 = DIAG1.*td;
B = find(~isfinite(diag1));
diag1(B) = zeros(size(B));


%diagonal sw --> ne
td2 = -sw./(ne-sw);
DIAG2 = (td2>=0 & td2<=1 & isfinite(td2));
diagflag2 = DIAG2;
diag2 = DIAG2.*td2;
B = find(~isfinite(diag2));
diag2(B) = zeros(size(B));

B = find(~isfinite(second));
second(B) = zeros(size(B));
B = find(~isfinite(first));
first(B) = zeros(size(B));

%Now, we combine the results from first, second, and diagonal points.

%The information is:
%firstflag, first
%secondflag, second
%diagflag, diag

end


levelindinv = ones(size(levelset))*-1;
levelindinv(levelind) = [1:length(levelind)];
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%matlab driver to perform subdomain solve
%
%Mike Flanagan 2002
%initial data is set through domainsetup.m
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%set initial values for f and g

f = complex(ones(1,N));
g = complex(zeros(1,N));


[levelind,levelset,firstflag, firstpoint, secondflag, secondpoint,...
 diagonalflag, diagonalpoint, diagonalflag2,diagonalpoint2] =...
                      newfulltriarea(P,M,N,k1,k2);
%if (sum(levelset1-levelset>0)>0)
%    disp('Proglem with levelsetconsitency')
%    pause
%end

[M N]

Pvec = reshape(P,1,(M+1)*N);
%K = mexcomputeintegral(top,bottom,M,N,levelind,levelset,firstflag,...
%firstpoint,secondflag,secondpoint,diagonalflag,diagonalpoint,Pvec,a1,a2);
%Klayered = buildlayer(K,L);
%disp('newfulltriarea completed inside domainsolve_noK')
%pause



[A,matIJ,RHS]=...
mexmakesparandrhs60(top,bottom,omega,alph,wKlayered,f,g,M,N,FOURIER,...
  levelind,levelset,firstflag,firstpoint,secondflag,...
  secondpoint,diagonalflag,...
  diagonalpoint,diagonalflag2,diagonalpoint2,Pvec,k1,k2,L);


%disp('Testing mexmakesparandrhs60')
%pause
SP = sparse(matIJ(:,1),matIJ(:,2),A);

finsolution = SP.'\RHS;

U = finsolution;

finaltop = finsolution((Mlayered+1)*N-N+1:(Mlayered+1)*N);
finaltrans = finsolution(1:N);

R = E0.*computefour([finaltop;finaltop(1)],N0);
```

```
R(1) = R(1)-exp(-im*2*b0(1)*y0);

T = E1.*computefour([finaltrans;finaltrans(1)],N1);

RU=R;
TU=T;


Fr = b0/b0(1).*(abs(R).^2);
Ft = b1/b0(1).*(abs(T).^2);


%subplot(2,1,2)
%imagesc(reshape(real(U),N,M+1))
```

```
function y = sparaction(X,temp,SP,M,N,top,bottom,omega,alpha,L,k1,k2)
% computes the action of our block-tridiagonal matrix
% without having to construct the matrix and do a matrix vector multiply
%temp is for use by matlab's gmres routine
%it has to pass a parameter into temp

     [m n] = size(X);


hy = (top-bottom)/M;

%[M N top bottom omega alpha L]


fillz = complex(zeros(N,1));

y = zeros(m,n);


%computing action
%FIRST LAYER

%-I    A12      0     g0
% 0    A22     -I     g1
%                     f2

u = X(N+1:2*N);
if (isreal(u)) u = complex(u);
end
[a b] = subsyscompute(SP,M,N,fillz,u,top,bottom,omega,alpha,k1,k2);

y(1:N) = -X(1:N) + a;
utemp = b - X(2*N+1:3*N);
%cant replace utemp yet because next layer will need u
%
% NOTE: the vector is being updated as we progress down the matrix
% multiply.
%


for k = 2:2:2*(L-2)
top = bottom;
bottom = bottom-hy;
u = X(k*N+1:(k+1)*N); v = X((k+1)*N+1:(k+2)*N);
if (isreal(u)) u = complex(u);
end
if (isreal(v)) v = complex(v);
end
[a b]= subsyscompute(SP,M,N,u,fillz,top,bottom,omega,alpha,k1,k2);
[c d]= subsyscompute(SP,M,N,fillz,v,top,bottom,omega,alpha,k1,k2);
```

```
y(k*N+1:(k+1)*N) = -X((k-1)*N+1:k*N) + a  + c;
y((k-1)*N+1:k*N) = utemp;
utemp = b + d  - X((k+2)*N+1:(k+3)*N);

end

%Fourth LAYER = Last LAYER for this example

u = X(2*(L-1)*N+1:(2*L-1)*N);
if (isreal(u)) u = complex(u);
end
top = bottom;
bottom = bottom -hy;
[a b] = subsyscompute(SP,M,N,u,fillz,top,bottom,omega,alpha,k1,k2);

utemp2 = -X((2*L-3)*N+1:2*(L-1)*N) + a ;
y((2*L-3)*N+1:2*(L-1)*N) = utemp;
y((2*L-2)*N+1:(2*L-1)*N) = utemp2;
y((2*L-1)*N+1:2*L*N) = b - X((2*L-1)*N+1:2*L*N);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
if (isreal(y))
     y = complex(y);
end

'complete sparaction'


%on specific layer we have certain floc and gloc
%compute A11 f
%        A21 f

%[Ut Ub]=subsyscompute(SP,M,N,floc,fillz,top,bottom,omega,alpha,K);


%compute A12 g
%        A22 g

%[Ut Ub]=subsyscompute(SP,M,N,fillz,gloc,....);
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Driver to compute tridiagonal subsystem solves
%
%
%Mike Flanagan, 2002
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

domainsetup

f = complex(ones(1,N));
g = complex(zeros(1,N));
Pvec = reshape(P,1,N*(M+1));
[levelind,levelset,firstflag, firstpoint, secondflag,...
         secondpoint, diagonalflag,diagonalpoint,diagonalflag2,...
         diagonalpoint2] = newfulltriarea(P,M,N,k1,k2);

[A,matIJ,RHS_nouse,Ltriangle] = ...
mexmakesparandrhs60_old(top,bottom,omega,alph,wKlayered,f,g,M,N,FOURIER,...
levelind,levelset,firstflag,firstpoint,secondflag,secondpoint,...
diagonalflag,diagonalpoint,diagonalflag2,diagonalpoint2,...
Pvec,k1,k2,1);

%(top,bottom,omega,alpha,K,M,N);
SubMat = sparse(matIJ(:,1),matIJ(:,2),A);
SubMat = SubMat.';
% in solving the system, it is necessary to compute transpose, but
% not conjugate transpose.
% the sparse matrix doesn't change from one layer to the next.

fillz = complex(zeros(size(ftop)));

%g0, g1, f2, g2, f3, g3, f4, ..., g(k), f(k+1), ... f(L), f(L+1)
% L is number of layers
% Vector X is the full vector of unknowns.

%f1  known
%------------------------
%g0


%f2
%------------------------
%g1


%f3
%------------------------
%g2
```

```
%fL
%------------------------
%gL-1


%fL+1
%------------------------
%gL   known


%operations to compute
%[ A11  A12] u   a
%[         ]   =
%[ A21  A22] v   b

%depending on choice of u and v



% RHS
% in gmres scheme the full matrix is computed via actions from
% subdomains
% RHS is the right hand side to be evaluated it is a vector length
% 2LN,  where L is number of layers and N is horizontal mesh size of domain
% Actually, it is nonzero for only  possibly  top 2N and bottom 2N components
% top incoming wave is given as ftop


rhs = complex(zeros(2*L*N,1));

%want A11 ftop
%want A21 ftop

%see paper for this description

[a b]=subsyscompute(SubMat,M,N,ftop,fillz,top,bottom,omega,alph,k1,k2);

rhs(1:2*N) = [-a;-b];

restrt = 15;
max_it = 100;
tol = 1e-6;

X = rhs;


%X = gmres('sparaction',rhs,restrt,tol,max_it,[],[],[],length(rhs),...
%SP,M,N,top,bottom,omega,alpha,K,L);
```

```
[X,error,iter,flag]=mygmres(SubMat,X,rhs,M,N,top,bottom,omega,alph,L,...
k1,k2,restrt,max_it,tol);

%y = sparaction(SP,X,M,N,top,bottom,omega,alpha,K,L);

F = []; temp = [];
for n = 1:length(rhs)
    a=zeros(length(rhs),1);
     a(n) = 1;
     y = sparaction(a,temp,SubMat,M,N,top,bottom,omega,alph,L,k1,k2);
F = [F;y.'];
end
F = F.';
%      matF = sprintf('fullmat%dx%dx%d',M,N,L);
%save matF F
%      sol = F\rhs

%      fname = sprintf('data%dx%dx%d',M,N,L);
%save fname X error iter flag

%'data saved'

%compare X with solutions

domainsolve

norm(X((2*L-1)*N+1:2*L*N)-finaltrans)
norm(X(1:N)-(finaltop-f.'))
```

```
function [ref, tran] = subsyscompute(SP,M,N,f,g,top,bottom,omega,alpha,k1,k2)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%algorithm computes the solution to our subdomain with incoming waves f and g
% SP exists prior to this function call
%
%
%Mike Flanagan, 2002
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
if (isreal(f))
     'f is not complex'
end
if (isreal(g))
'g is not complex'
end


RHS = mexmakerhs(top,bottom,omega,alpha,f,g,M,N,k1,k2);


solution = SP\RHS;

%whos f
%whos g
%whos solution
ref = solution(M*N+1:(M+1)*N)-f;
tran = solution(1:N)-g;

%should return a column vector.
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Routine computes adjoint solve to generate gradient
%
% Mike Flanagan, 2002
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


%set initial values for f and g
f = complex(ones(1,N));
g = complex(zeros(1,N));

[M N]

Pvec = reshape(P,1,(M+1)*N);

[A,matIJ,RHS,lefttriangle]=...
mexmakesparandrhs60(top,bottom,omega,alph,wKlayered,f,g,M,N,FOURIER,...
                    levelind,levelset,firstflag,firstpoint,secondflag,...
                    secondpoint,diagonalflag,diagonalpoint,...
                    diagonalflag2,diagonalpoint2,...
                    Pvec,k1,k2,L);

%disp('came out of mexmakesparandrhs60')
%pause


SP = sparse(matIJ(:,1),matIJ(:,2),A);
finsolution = SP.'\RHS;
U = finsolution;


finaltop = finsolution((Mlayered+1)*N-N+1:(Mlayered+1)*N);
finaltrans = finsolution(1:N);

%R = vector of reflection coefficients.
%For each mode we compute by fourier.

R = E0.*computefour([finaltop;finaltop(1)],N0);
R(1) = R(1)-exp(-im*2*b0(1)*y0);

T = E1.*computefour([finaltrans;finaltrans(1)],N1);
RU = R;
TU = T;
vec0 = 2*b0/b0(1).*((b0/b0(1).*abs(R).^2-abs(targetreflect).^2).*R...
.*conj(E0))/(2*pi);
vec1 = 2*b1/b0(1).*((b1/b0(1).*abs(T).^2-abs(targettrans).^2).*T.*...
```

```
conj(E1))/(2*pi);


%%%Keep in mind that the vector dotproduct depicted above is for the
%%%variational integral setup in the function call below.
%%%The PDE boundary conditions have the negative of the above.



if (isreal(vec0))
 vec0 = complex(vec0);
end
if (isreal(vec1))
vec1 = complex(vec1);
end



RHSADJ = mexmakerhsadjoint60(top,bottomlayered,omega,alph,...
wKlayered,Mlayered,N,vec0,N0,length(N0),vec1,N1,length(N1),FOURIER);

%disp('came out of mexmakerhsadjoint60');
%pause
[SP2, matij]=mexmakesparadjoint60(top,bottom,omega,alph,wKlayered,f,g,...
M,N,FOURIER,levelind,levelset,firstflag,firstpoint,secondflag,secondpoint,...
diagonalflag,diagonalpoint,diagonalflag2,diagonalpoint2,...
Pvec,k1,k2,L);


SPadj = sparse(matij(:,1),matij(:,2),SP2);

W = SPadj.'\RHSADJ;
```

VITA

Michael Brady Flanagan was born in Mt. Vernon, Washington on January 20, 1970. Mike went to high-school in Southern California. He received his Bachelor's degree from Montana State University in 1992, and a Master's degree from University of Alaska-Fairbanks in 1995. He has spent 2 summers at Lawerence Livermore National Lab working on thesis related research. His recreational interests include hiking/backpacking and fishing. Mike's permanent address is 2125 White Lane, Dillon, Mt, 59725

The typist for this thesis was Mike Flanagan.